LOW-RANK ESTIMATION AND EMBEDDING LEARNING: THEORY AND APPLICATIONS

BY

HUAN GUI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Doctoral Committee:

Professor Jiawei Han, Chair
Assistant Professor Jian Peng
Professor ChengXiang Zhai
Doctor Cong Yu, Google Research

Abstract

In many real-world applications of data mining, datasets can be represented using matrices, where rows of the matrix correspond to objects (or data instances) and columns to features (or attributes). Often the datasets are in high-dimensional feature space. For example, in the vector space model of text data, the feature dimension is the vocabulary size. If representing a social network using an adjacency matrix, the feature dimension corresponds to the number of objects in the network. Many other datasets also fall into this category, such as genetic datasets, images, and medical datasets. Even though the feature dimension is enormous, a common observation is that the high-dimensional datasets may (approximately) lie in a subspace of smaller dimensionality, due to dependency or correlation among features. This thesis studies the problem of automatically identifying the low-dimensional space that high-dimensional datasets (approximately) lie in based on dimension reduction models: one is low-rank estimation models and the other is embedding learning models. For data matrices, low-rank estimation is to recover an underlying data matrix, subject to the constraint the matrix is of reduced rank. Such analysis is also generalized to the high-dimensional higher-order tensor data. Meanwhile, embedding learning models are to directly project the observation data into a low-dimensional vector space.

In the first part, the theoretical analysis of low-rank estimation models is established in the regime of high-dimensional statistics. For matrices, the low-rank structure corresponds to the sparsity of the singular values; while for tensors, the low-rank model can be defined as the low-rankness of the unfolding matrices of the tensor. To achieve low-rank solutions, two categories of regularization are imposed. Firstly, the problem of robust tensor decomposition with gross corruption is considered. To recover the underlying true tensor and corruption of large magnitude, structure assumptions of low-rankness and sparsity are imposed on the tensor and corruption, respectively. The Schatten-1 norm is applied as convex regularization for the low-rank structure. Secondly, the problem of matrix estimation is considered with a nonconvex penalty. Compared with convex regularization, nonconvex penalty takes advantage of the large singular values, which leads to faster statistical convergence rate and oracle property under a mild condition on the magnitude of the singular values. For both problems, efficient optimization algorithms are proposed, and extensive numerical

experiments are conducted to corroborate the efficacy of the proposed algorithms and the theoretical analysis.

In the second part, embedding learning models for real world applications are presented. The high-dimensional data is projected into a low-dimensional vector space via preserving the proximity among objects. Each object is represented by a low-dimensional vector, called embedding or distributed representation. In the first application, the heterogeneity of the objects is considered. Based on the observation that several interactions among the strongly-typed objects happen simultaneously as an event, the embeddings of objects in each event are learned as a whole. In other words, the model preserves the proximity among all the participating objects in each event. Experimental results provide evidence that the learned embeddings are more effective while being robust to data sparsity and noises for various classification tasks. In the second application, the task of expert finding is studied, which is to rank candidates with appropriate expertise based on a given query. To capture the subtle semantic information regarding specific queries with narrow semantic meanings, locally-trained embedding learning with concept hierarchy as guidance is proposed for query expansion. The locally-trained embeddings preserve the proximity among terms constrained on a sub-corpus. Compared with global embedding trained on the whole dataset, locally-trained embedding has stronger representation power. Experimental results show that the proposed embedding learning method achieves high precision regarding the task of expert finding.

To summarize, this thesis provides important results of low-rank estimation and embedding learning models for high-dimensional data analysis and real-world applications.

To my family, for their love and support.

Acknowledgments

I would like to thank all the people and agencies who give me tremendous support and help to make this thesis happen.

First of all, I am deeply grateful to my advisor, Professor Jiawei Han. Your extraordinary expertise and keen insights in data mining and related areas, your passion and dedication for scientific research, your patience in guiding students, and your kindness and wisdom have been motivating and inspiring me through my Ph.D. study. I always feel extremely fortunate and thankful to have you as my advisor, for all the insightful discussions, earnest encouragements, and strongest supports in every means. Without your support and openness, this thesis would never have been possible.

I greatly appreciate the guidance and support from my thesis committee members, Professor Chengxiang Zhai, Professor Jian Peng, and Doctor Cong Yu. Professor ChengXiang Zhai has provided insightful suggestions and invaluable support to many of my research projects. Professor Jian Peng has given me many detailed comments on my thesis and helped make some sense of the confusion. I am thankful to Doctor Cong Yu for not only serving on my thesis committee, but also mentoring me through a marvelous internship at Google Research, which broadened my vision on both research and real-world products. Thank you for your invaluable comments and advice on my research and thesis work.

I have spent great summers at LinkedIn and Google. These experiences not only taught me how to do research with a tight schedule, but also gave a chance to understand and solve industrial problems, which are fun with lots of implications. I am grateful for all my mentors, Haishan Liu, Ya Xu, Cecilia Chen, and Cong Yu, for the enormous enthusiasm and support. Working with you taught me to ask different questions and think differently. Ya is a great collaborator and supportive friend. Thank you for insights for research and the advice for life.

I have had the great pleasure and honor of working with other coauthors and collaborators, including Anmol Bhasin, George Brova, Quanquan Gu, Meng Jiang, Lance Kaplan, Jialu Liu, Liyuan Liu, Xiangrui Meng, Brandon Norick, Meng Qu, Yu Shi, Yizhou Sun, Fangbo Tao, Aston Zhang, Qi Zhu. I owe my deepest gratitude to them. Especially, thank Yizhou for guiding me through my first research projects and

everything that followed. I learned a lot from Quanquan and Jialu. Working with you was always fun and inspiring.

I would like to thank all my friends and my fellows from Data Mining Group, Database and Information System Group, and the Computer Science at University of Illinois at Urbana Champaign. Without you, life would not be the same. Especially, I want to thank Jim Cai, Po-wei Chan, Mingcheng Chen, Ahmed El-Kishky, Kuang Gong, Wenqi He, Shan Jiang, Liang Liu, Stephen Macke, Haoruo Peng, Kent Quanrud, Zhenhui Li, Xiang Ren, Jingbo Shang, Lu Su, Juer Song, Wenzhu Tong, Chi Wang, Jingjing Wang, Xiaolong Wang, Yiren Wang, Doris Xin, Haoyi Xu, Quan Yuan, Chao Zhang, Keyang Zhang, Yinan Zhang, Shi Zhi, Rongda Zhu, Honglei Zhuang. Thank you for all the happy moments.

Finally, I thank my parents, brother, and sister, for endless love, encouragement, understanding, and support. Without you, I would never achieve what I have now. 谁言寸草心，报得三春晖。

Table of Contents

Chapter 1 Introduction **1**
1.1 Background and Motivation 1
1.2 Low-rank Estimation Models 2
1.2.1 Low-rank Decomposition Models via Convex Regularizations 3
1.2.2 Low-rank Estimation Models via Nonconvex Penalty 5
1.3 Embedding Learning Models 6
1.3.1 Embedding Learning with Events for Classification 7
1.3.2 Locally-trained Embedding Learning for Expert Finding 9
1.4 Notation 12
1.5 Organization of the Thesis 13

Chapter 2 Literature Review **14**
2.1 Low-rank Estimation Models 14
2.1.1 Low-rank Estimation Models with Convex Regularization 14
2.1.2 Low-rank Estimation Models with Nonconvex Penalty 15
2.2 Embedding Learning Models 16
2.2.1 Embedding Learning in Heterogeneous Information Networks 16
2.2.2 Embedding Learning for Expert Finding 17

Chapter 3 Low-rank Decomposition Models with Convex Regularization **19**
3.1 Notation and Background for Tensors 19
3.2 Main Results 21
3.2.1 Deterministic Bounds 21
3.2.2 Noisy Tensor Decomposition 24
3.3 Algorithm 25
3.4 Experiments 26

Chapter 4 Low-Rank Estimation Models with Nonconvex Penalty **28**
4.1 Low-rank Matrix Estimation with Nonconvex Penalty 28
4.1.1 The Observation Model 28
4.1.2 Examples 28
4.1.3 The Proposed Estimator 29
4.1.4 Optimization Algorithm 30
4.2 Main Theory 32
4.2.1 Results for the Generic Observation Model 34
4.2.2 Results for Specific Examples 36
4.3 Numerical Experiments 38
4.3.1 Simulations 38
4.3.2 Experiments on Real World Datasets 40

Chapter 5 Embedding Learning in Heterogeneous Information Networks with Events . 41
5.1 Preliminaries . . . 41
5.1.1 Heterogeneous Information Networks and Events . . . 41
5.1.2 Learning Object Embedding . . . 42
5.2 HEBE Framework . . . 44
5.2.1 SubEvent Sampling . . . 44
5.2.2 HEBE Object Prediction . . . 45
5.2.3 HEBE Hyperedge Prediction . . . 47
5.2.4 Multiple Event Types . . . 49
5.3 Optimization . . . 49
5.3.1 Noise Pairwise Ranking . . . 49
5.3.2 Optimization for HEBE-PO . . . 51
5.3.3 Optimization for HEBE-PE . . . 53
5.3.4 Unified Algorithm . . . 54
5.4 Experimental Study . . . 54
5.4.1 Datasets and Compared Methods . . . 55
5.4.2 Evaluation Metric . . . 56
5.4.3 Experimental Results . . . 58
5.4.4 Model Study . . . 61

Chapter 6 Locally-trained Embedding Learning For Expert Finding . . . 65
6.1 Preliminary . . . 65
6.1.1 Heterogeneous Bibliographical Networks . . . 65
6.1.2 The Document-based Models . . . 65
6.1.3 Word Embedding Learning . . . 66
6.2 Local Embedding via Concept Hierarchy . . . 67
6.2.1 Concept Hierarchy . . . 67
6.2.2 Locally-trained Embedding Learning . . . 68
6.3 Expert Ranking in Relevance Network . . . 70
6.3.1 Relevance Network Construction . . . 70
6.3.2 Ranking in Relevance Network . . . 70
6.4 Experimental Results . . . 71
6.4.1 Experimental Setup . . . 72
6.4.2 Experimental Results . . . 73

Chapter 7 Conclusions and Future Work . . . 76

Bibliography . . . 78

Appendix A Proof of Chapter 3 . . . 85
A.1 Proof of Lemma 3.2.1 . . . 85
A.2 Proof of Theorem 3.2.2 . . . 86
A.3 Proof of Theorem 3.2.4 . . . 88
A.4 Proof of Corollary 3.2.5 . . . 89
A.5 Proof of Lemma 3.2.6 . . . 91

Appendix B Proof of Chapter 4 . . . 92
B.1 Introduction . . . 92
B.2 Additional Experimental Results . . . 92
B.3 Background . . . 92
B.4 Proof of the Main Results . . . 94
B.4.1 Proof of Theorem 4.2.4 . . . 94
B.4.2 Proof of Theorem 4.2.5 . . . 100
B.5 Proof of the Results for Specific Examples . . . 104

B.5.1 Matrix Completion . . . 105
B.5.2 Matrix Sensing With Dependent Sampling . . . 109
B.6 Proof of Auxiliary Lemmas . . . 111
B.6.1 Proof of Lemma B.4.1 . . . 112
B.6.2 Proof of Lemma B.4.2 . . . 113
B.6.3 Proof of Lemma B.4.3 . . . 114
B.6.4 Proof of Lemma B.5.3 . . . 115

Chapter 1

Introduction

1.1 Background and Motivation

In many real-world applications of data mining, datasets can be represented using matrices, where rows in the matrix correspond to objects (or data instances) and columns to features (or attributes). Often the datasets are in high-dimensional feature space. However, a key observation is that even though the data is of high dimension, there is likely to be dependency or correlation among features. In other words, there is redundancy in the large feature space and the dataset may (approximately) lie in a subspace of smaller dimensionality. Consider the following high-dimensional data examples.

Example 1.1.1 (Text Data)**.** *Given a collection of documents, it can be represented as a matrix using the vector space model. In the matrix, each document corresponds to a row; each column a word in the vocabulary; and values denote the frequency of the corresponding word in the corresponding document. Therefore, the feature space is the vocabulary. Since the size of the vocabulary is gigantic, the text data is in high-dimensional feature space. However, many words in the vocabulary share identical or similar meanings, resulting in redundancy in the feature space.*

Example 1.1.2 (Recommender System)**.** *In a recommendation system, users give ratings to items they have purchased. Therefore, the ratings can be represented by a matrix, where rows correspond to users while columns as items. Values in the matrix denote the ratings from users to items, accordingly. The feature space is the set of all available items. Generally, since cardinality of the item set is huge, the rating data is of high dimension. However, many movies belong to the same genres and share similar characteristics, which results in redundancy in the feature space.*

In the high-dimensional dataset, the redundancy in the feature space (as shown in Example 1.1.1 and Example 1.1.2) implies that the data (approximately) lies in a lower-dimensional space. In Example 1.1.1, topic models project the text data into a low-dimensional space, defined by distinct topics; for Example 1.1.2, collaborative filtering methods project the rating data into a low-dimensional space, defined by latent pref-

erence factors. Motivated by the examples, a natural question arises: how to automatically identify the lower-dimensional space that the high-dimensional data (approximately) lie in?

This thesis seeks answers to this question with dimension reduction models. The first approach is low-rank estimation models. Regarding matrices, the rank is defined as the dimension of the vector space that spanned by its columns, which is the same as the dimension of the vector space that spanned by its rows. The low-rank assumption is to reduce the dimension of the corresponding vector space. Embedding learning is to directly project a new low-dimensional vector space via preserving the proximity among objects in the observation dataset. Many embedding learning models can be interpreted as generalized low-rank models [99].

1.2 Low-rank Estimation Models

Low-rank matrix estimation model is to fit the observation data with a matrix with a distance measure, subject to the constraint that the matrix is low-rank. The rank of a matrix corresponds to the dimension of the vector space that its columns span, which is also equivalent to the number of nonzero singular values. Therefore, the low-rankness can be imposed as the sparsity of singular values.

Suppose the data matrix is $\mathbf{M} \in \mathbb{R}^{m_1 \times m_2}$, Ω is a subset of entries being observed with noise, and $\mathbf{\Theta}$ is the low-rank matrix that approxiamtes $\mathbf{M}$, $\mathbf{\Theta}$ can be estimated via solving the following optimization problem [17, 18]:

$$\widehat{\mathbf{\Theta}} = \underset{\mathbf{\Theta} \in \mathbb{R}^{m_1 \times m_2}}{\operatorname{argmin}} \left\| \mathcal{Z}_\Omega(\mathbf{M}) - \mathcal{Z}_\Omega(\mathbf{\Theta}) \right\|_F + \mathcal{R}_\lambda(\mathbf{\Theta}), \tag{1.2.1}$$

where $\mathcal{Z}_\Omega$ is the sampling operator regarding Ω, $\|\cdot\|_F$ is the Frobenius norm (i.e., the squared root of the sum of squared singular values), and $\mathcal{R}_\lambda$ is the regularization with parameter λ to encourage the sparsity of $\mathbf{\Theta}$'s singular values (i.e., low-rankness of $\mathbf{\Theta}$).

Estimation of low-rank matrices [86, 19, 82, 53, 18, 43, 36, 42] has received increasing interest in the past decade. It has broad applications in many fields such as data mining and computer vision. One prominent example of matrix estimation is matrix completion, which is to recover the underlying matrix based on partial observations of entries in the matrix with noise. For example, in the recommendation systems, one aims to predict the unknown preferences of a set of users over a set of items, provided a partially observed rating matrix. Another application of low-rank matrix estimation is image inpainting, to recover missing pixels based on a portion of pixels being observed.

1.2.1 Low-rank Decomposition Models via Convex Regularizations

In this section, the low-rank model with convex regularization is discussed. More concretely, in (1.2.1), we have $\mathcal{R}_\lambda(\mathbf{\Theta})$ as a convex function. A popular choice of $\mathcal{R}_\lambda(\mathbf{\Theta})$ is the nuclear norm [86, 19, 79, 71, 53], $\mathcal{R}_\lambda(\mathbf{\Theta}) = \lambda\|\mathbf{\Theta}\|_*$. Therefore, (1.2.1) can be written as

$$\widehat{\mathbf{\Theta}} = \underset{\mathbf{\Theta}\in\mathbb{R}^{m_1\times m_2}}{\operatorname{argmin}} \left\|\mathcal{Z}_\Omega(\mathbf{M}) - \mathcal{Z}_\Omega(\mathbf{\Theta})\right\|_F + \lambda\|\mathbf{\Theta}\|_*. \tag{1.2.2}$$

It is worth noting that besides nuclear norm, there are many other rank proxy functions, such as Schatten-p norm [82, 74], max norm [87, 15], the von Neumann entropy [52]. Nuclear norm is probably the most widely used since it is the tightest convex relaxation of the matrix rank.

Moreover, the low-rank model via convex regularization (1.2.2) is generalized to tensor data. Tensor data analysis has witnessed increasing applications in machine learning, data mining, and computer vision. For example, an ensemble of face images can be modeled as a tensor, whose mode corresponds to pixels, subjects, illumination, and viewpoint [102].

Firstly, we consider the low-rank modeling of tensor data. Traditional tensor decomposition methods such as Tucker decomposition and CANDECOMP/PARAFAC(CP) decomposition [56, 50] aim to factorize an input tensor into a number of low-rank factors. However, they are prone to local optima because they are solving essentially non-convex optimization problems. In order to address this problem, [62] [96] extended the nuclear norm of matrices [88] to tensors, and generalized convex matrix completion [19] [18] and matrix decomposition [17] to convex tensor completion/decomposition. For example, the goal of tensor decomposition aims to accurately estimate a low-rank tensor $\mathcal{W} \in \mathbb{R}^{n_1\times\cdots\times n_K}$ from the noisy observation tensor $\mathcal{Y} \in \mathbb{R}^{n_1\times\cdots\times n_K}$ that is contaminated by dense noises, i.e., $\mathcal{Y} = \mathcal{W}^* + \mathcal{E}$, where $\mathcal{W}^* \in \mathbb{R}^{n_1\times\cdots\times n_K}$ is a low-rank tensor, $\mathcal{E} \in \mathbb{R}^{n_1\times\cdots\times n_K}$ is a noise tensor whose entries are i.i.d. Gaussian noise with zero mean and bounded variance σ^2, i.e., $\mathcal{E}_{i_1,\ldots,i_K} \sim N(0,\sigma^2)$. [98] [97] analyzed the statistical performance of convex tensor decomposition under different extensions of nuclear norm. They showed that, under certain conditions, the estimation error scales with the rank of the true tensor $\mathcal{W}^*$. Furthermore, they demonstrated that given a noisy tensor, the true low-rank tensor can be recovered under restricted strong convexity assumption [73]. However, all these algorithms [62] [96] and theoretical results [98] [97] reply on the assumption that the observation noise has a bounded variance σ^2. Without this assumption, we are not able to identify the rank of $\mathcal{W}^*$, and therefore the estimated low-rank tensor $\widehat{\mathcal{W}}$ could be very far from the true tensor $\mathcal{W}^*$.

On the other hand, in many practical applications such as face recognition and image/video denoising, a portion of the observation tensor $\mathcal{Y}$ might be contaminated by gross error due to illumination, occlusion or

pepper/salt noise. This scenario is not covered by finite variance noise assumption, therefore new mathematical models are demanded to address this problem. This motivates us to study convex tensor decomposition with gross corruption. It is clear that if all the entries of a tensor are corrupted by large error, there is no hope to recover the underlying low-rank tensor. To overcome this problem, one common assumption is that the gross corruption is sparse. Under this assumption, together with previous low-rank assumption, we formalize the noisy linear observation model as follows:

$$\mathcal{Y} = \mathcal{W}^* + \mathcal{V}^* + \mathcal{E}, \tag{1.2.3}$$

where $\mathcal{W}^* \in \mathbb{R}^{n_1 \times \ldots \times n_K}$ is a low-rank tensor, $\mathcal{V}^* \in \mathbb{R}^{n_1 \times \ldots \times n_K}$ is a sparse corruption tensor, where the locations of nonzero entries are unknown and the magnitudes of the nonzero entries can be arbitrarily large, and $\mathcal{E} \in \mathbb{R}^{n_1 \times \ldots \times n_K}$ is a noise tensor whose entries are i.i.d. Gaussian noise with zero mean and bounded variance σ^2, and thus dense. Our goal is to recover the low-rank tensor $\mathcal{W}^*$, as well as the sparse corruption tensor $\mathcal{V}^*$. Note that in some applications, the corruption tensor is of independent interest and needs to be recovered.

Given the observation model in (1.2.3), and the low-rank as well as sparse assumptions on $\mathcal{W}^*$ and $\mathcal{E}^*$ respectively, we propose the following convex minimization to estimate the unknown low-rank tensor $\mathcal{W}^*$ and the sparse corruption tensor $\mathcal{E}^*$ simultaneously:

$$\arg\min_{\mathcal{W},\mathcal{V}} |||\mathcal{Y} - \mathcal{W} - \mathcal{V}|||_F^2 + \lambda_M |||\mathcal{W}|||_{S_1} + \mu_M |||\mathcal{E}|||_1 , \tag{1.2.4}$$

where $|||\cdot|||_{S_1}$ is tensor Schatten-1 norm [98], $|||\cdot|||_1$ is entry-wise ℓ_1 norm of tensors, and λ_M and μ_M are positive regularization parameters. We call this optimization *Robust Tensor Decomposition*, which can been seen as a generalization of convex tensor decomposition in [62] [96] [98]. The regularization associated with the $\mathcal{V}$ encourages sparsity on the corruption tensor, where parameter μ_M controls the sparsity level. In this thesis, we study various conditions for the size of the tensor, the rank of the tensor, and the fraction (sparsity level) of the corruption, such that (1.2.4) is able to recover $\mathcal{W}^*$ and $\mathcal{V}^*$ with small estimator error and (1.2.4) is able to recover the exact rank of $\mathcal{W}^*$ and the support of $\mathcal{V}^*$.

In Chapter 3, we focus on the following questions: under what conditions for the size of the tensor, the rank of the tensor, and the fraction (sparsity level) of the corruption so that: (i) (1.2.4) is able to recover $\mathcal{W}^*$ and $\mathcal{V}^*$ with small estimator error? (ii) (1.2.4) is able to recover the exact rank of $\mathcal{W}^*$ and the support of $\mathcal{V}^*$? we present nonasymptotic error bounds in high-dimensional statistics to answer these questions. Experiments on synthetic datasets validate our theoretical results.

1.2.2 Low-rank Estimation Models via Nonconvex Penalty

Considering the low-rank estimation model with convex penalty, (1.2.2) is to impose the low-rank structure via nuclear norm, which is defined as the ℓ_1 penalty on the singular values of $\mathbf{\Theta}$. Meanwhile, it is now well-known that ℓ_1 penalty in Lasso [28, 120, 125] introduces a bias into the resulting estimator, which compromises the estimation accuracy. In contrast, nonconvex penalties such as smoothly clipped absolute deviation (SCAD) penalty [28] and minimax concave penalty (MCP) [120] are favored in terms of estimation accuracy and variable selection consistency [110]. Due to the close connection between ℓ_1 norm and nuclear norm (nuclear norm can be seen as an ℓ_1 norm defined on the singular values of a matrix), nonconvex penalties for low-rank matrix estimation have recently received increasing attention for low-rank matrix estimation. Typical examples of nonconvex approximation of the matrix rank include Schatten ℓ_p-norm $(0 < p < 1)$ [74], the truncated nuclear norm [39], and the MCP penalty defined on the singular values of a matrix [108, 61]. Although good empirical results have been observed in these studies [74, 39, 108, 61, 66, 117], little is known about the theory of nonconvex penalty for low-rank matrix estimation. In this thesis, we will provide theoretical analysis of low-rank estimation with nonconvex penalty.

Hence, a simple model for low-rank matrix completion is introduced. The estimator of a more general observation model for matrix estimation will be discussed in Chapter 4. The low-rank estimation model with nonconvex penalty for matrix completion is proposed as follows:

$$\widehat{\mathbf{\Theta}} = \underset{\mathbf{\Theta}\in\mathbb{R}^{m_1\times m_2}}{\operatorname{argmin}} \left\|\mathcal{Z}_\Omega(\mathbf{M}) - \mathcal{Z}_\Omega(\mathbf{\Theta})\right\|_F + \mathcal{P}_\lambda(\mathbf{\Theta}), \tag{1.2.5}$$

where $\mathcal{P}_\lambda(\cdot)$ is a nonconvex function applied to the singular values of $\mathbf{\Theta}$. We look at two examples of nonconvex penalties (*e.g.*, SCAD and MCP) and compare them with convex regularization (ℓ_1 norm) in Figure 1.1.

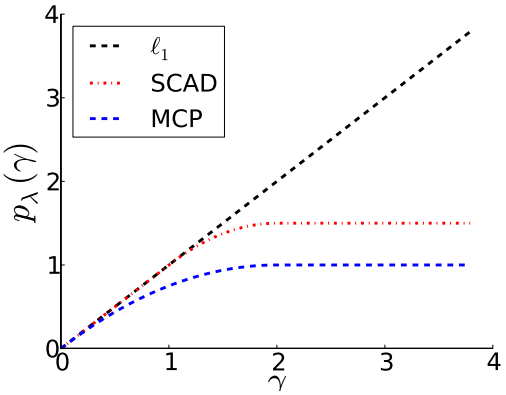


Figure 1.1: Comparison of convex regularization and nonconvex penalty.

As shown in Figure 1.1, where γ denotes the magnitude of a singular value, ℓ_1 norm (corresponding to nuclear norm) linearly increases when γ increases, while nonconvex penalties remain the same after γ exceeds some threshold. Recall that both nuclear norm and nonconvex penalties serve as surrogate functions for matrix rank. Therefore, convex penalty overpenalizes singular values with large magnitude; while nonconvex penalty does not.

In Chapter 4, it is rigorously shown that the low-rank estimation model with nonconvex penalty, by taking advantage of singular values with large magnitude, attains faster statistical convergence rates, compared with the conventional estimator with nuclear norm penalty. Furthermore, under a mild assumption on the magnitude of the singular values, the proposed estimator enjoys oracle property; that is exactly recovering the true rank of the underlying matrix, as well as attains a faster rate.

1.3 Embedding Learning Models

In Section 1.2, the low-rank model is discussed as in (1.2.1). In this section, embedding learning is introduced for large-scale information network analysis. Embedding learning is to directly project the objects in the network into a low-dimensional vector space and present each object using a low-dimensional vector. The objective of embedding learning is preserve the proximity among objects based on the observations.

Embedding learning is an important task in unsupervised learning and in data preprocessing of supervised learning. The low-dimensional vectors, as distributed representations of the relationships among objects, are beneficial for various downstream applications, such as exploratory data analysis, link prediction [60], visualization [100], object clustering [76], classification [7], and recommendation [54]. The objective of embedding techniques is mainly to preserve certain relationships among objects [30, 68, 78, 21, 33, 93, 92, 55, 4, 94].

Principal component analysis (PCA) [48, 104, 99] is a classical embedding learning model based on the low-rank structure. PCA approximates the observation matrix $\mathbf{M}$ with the best rank-k matrix ($k \ll \min\{m_1, m_2\}$) via solving the optimization problem

$$\text{minimize} \quad \|\mathbf{M} - \mathbf{\Theta}\|_F^2, \tag{1.3.1}$$

$$\text{subject to} \quad \text{rank}(\mathbf{\Theta}) \leq k. \tag{1.3.2}$$

Unlike (1.2.1), which impose the low-rankness via the regularization, the rank of $\mathbf{\Theta}$ can be encoded via a

factorization form such that $\mathbf{\Theta} = \mathbf{U}^\top \mathbf{V}$, where $\mathbf{U} \in \mathbb{R}^{m_1 \times k}$ and $\mathbf{V} \in \mathbb{R}^{m_2 \times k}$. Thus, (1.3.1) is equivalent as

$$\text{minimize} \quad \|\mathbf{M} - \mathbf{U}^\top \mathbf{V}\|_F^2. \tag{1.3.3}$$

If $\mathbf{U} = [\mathbf{u}_1^\top, \mathbf{u}_2^\top, \ldots, \mathbf{u}_{m_1}^\top]^\top$, $\mathbf{u}_i$ is the i-th row of $\mathbf{U}$; similarly, $\mathbf{v}_j$ is defined as j-th row of $\mathbf{V}$. $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{R}^k$ correspond to the embeddings of the corresponding objects or features. Moreover, we have that (1.3.3) can be rewritten as

$$\text{minimize} \quad \left(M_{i,j} - \mathbf{u}_i^\top \mathbf{v}_j\right)^2, \tag{1.3.4}$$

where $M_{i,j}$ is the (i, j) element of $\mathbf{M}$. In the applications information network analysis, where $\mathbf{M}$ denotes the weighted adjacency matrix, $M_{i,j}$ denotes the edge weight between object i and j. Therefore, the learned embeddings $\mathbf{u}_i$'s, $\mathbf{u}_j$'s for $1 \leq i \leq m_1, 1 \leq j \leq m_2$, are the preserve the tie strength in the network as in (1.3.4). For large-scale information network analysis, different proximity to be preserved for embedding learning under different scenario is explored, as we will discuss in Chapter 5 and Chapter 6 for concrete applications.

1.3.1 Embedding Learning with Events for Classification

Embedding learning with strongly-typed interactions has broad real-world applications [92, 21]. In (1.3.4), the embeddings are obtained via preserving the tie-strength between objects. Here, a new framework is introduced for embedding learning in heterogeneous information networks. We use DBLP as an illustration example.

Example 1.3.1. *DBLP (http://dblp.uni-trier.de) is a CS bibliographical dataset, where each publication record corresponds to an event. There are three types of participating objects: authors (A), terms (T), and venue (V), with their interactions represented at the schema level as shown in Fig. 1.2 (left). To learn object embeddings, we need to preserve the proximity among all the participating objects (Fig. 1.2, top right). Previous studies (e.g., [92, 21]) decompose the simultaneous interaction among all objects into several scattered pairwise interactions (e.g., Author-Paper, Venue-Paper), as shown in Fig. 1.2 (bottom right). Object embeddings are then learned by combing embedding learning procedures upon each set of pairwise interactions, using existing embedding learning methodologies developed for single-typed network data. However, such pairwise interactions may miss some important information. Consider Einstein and Hawking may publish in the same venue, using similar terms in astrophysics, but they did not coauthor a paper. Pairwise*

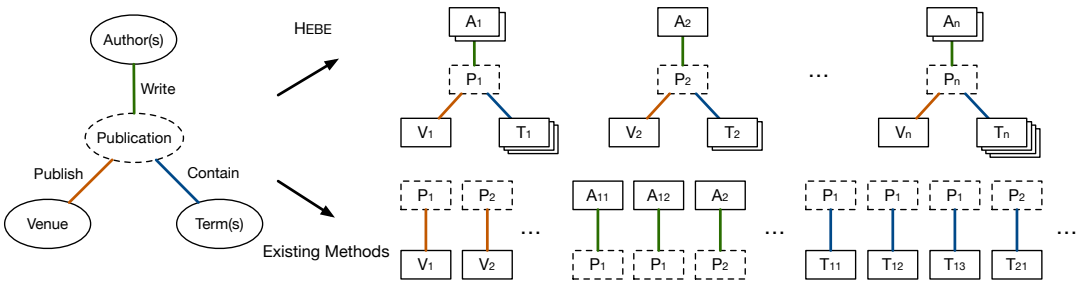


Figure 1.2: The interaction schema of DBLP is in the left. A publication event results in the interactions of authors-publication, venue-publication, and terms-publication at the same. Existing methods (in the bottom right) consider each interaction type independently. Our method (in the top right) defines the set of interactions resulted from the same event as a hyperedge, and model each hyperedge as a whole.

modeling cannot capture such subtle differences.

To learn object embeddings, we need to preserve the proximity among all the participating objects (Fig. 1.2, top right). Previous studies (e.g., [92, 21]) decompose the simultaneous interaction among all objects into several scattered pairwise interactions (e.g., Author-Paper, Venue-Paper), as shown in Fig. 1.2 (bottom right). Object embeddings are then learned by combing embedding learning procedures upon each set of pairwise interactions, using existing embedding learning methodologies developed for single-typed network data. However, such pairwise interactions may miss some important information. Consider Einstein and Hawking may publish in the same venue, using similar terms in astrophysics, but they did not coauthor a paper. Pairwise modeling cannot capture such subtle differences.

We propose a generic framework called **H**yper**E**dge **B**ased **E**mbedding (HEBE) that captures multiple interactions at the same time, as illustrated in the top right of Figure 1.2. Inspired from classical hypergraph theory [6] on hyperedges, we define the set of participating objects with interactions happening simultaneously as a *hyperedge*. It is worth noting that the hyperedge defined here is more general than it is defined in classical hypergraph theory since we consider the participating objects might be of different types. For each event, we model the proximity of the interaction among all participating objects in the hyperedge as a whole. Hyperedges provide us with a more complete description of events, therefore our methods preserves more contextual information for embedding learning in heterogeneous information networks.

We first propose two methods based on different prediction semantics to model the proximity of each event. The first method HEBE-**P**redict **O**bject (HEBE-PO) is to predict *if a participating object (as target) would be observed in an event given all the other participating objects.* This method is based on the observation that all the participating objects in an event share semantic similarity. Our second HEBE-**P**redict Hyper**E**dge (HEBE-PE) is to predict *if the hyperedge can be observed given all the participating objects.*

Similar to HEBE-PO, HEBE-PE is based on the observation of semantic similarity among participating objects. The distance measure is defined as KL divergence between empirical distribution and parameterized conditional probability distribution.

Since HEBE is applied to high-dimensional data, we leverage recent advancement of asynchronous stochastic optimization [80] to take advantage of the parameter sparsity in high-dimensional data. Furthermore, we devise a new optimization technique, called *Noise Pairwise Ranking*, on the conditional probability of prediction. In sharp comparison with the existing methods, our method is free of negative sampling hyperparameter [68, 93, 69].

In HEBE, each hyperedge encapsulates more contextual information, leading to more informative and efficient updates. Consequently, HEBE is more robust to data sparseness and noise. We apply HEBE to real-world high-dimensional datasets to learn object embeddings and measure the quality of the learned embeddings based on various classification tasks. We observe that HEBE produces embeddings with better classification accuracy results while being robust to data sparseness and noise.

1.3.2 Locally-trained Embedding Learning for Expert Finding

For a project on "information extraction", who would be able to provide guidelines for problem solving? For a new funding proposal on "ontology alignment", who would be able to review and make good assessment? For the upcoming PKDD conference on "data mining", who should be invited to give a keynote speech? *Experts.*

Expert finding [3, 26, 107, 123] is defined as the problem of ranking the candidate with appropriate expertise for a given query. The problem receives increasing attention in academia due to the TREC Expert Finding Track [85]. Accurate candidate ranking has broad applications. However, the problem is particularly challenging since a query can be as *general* as "data mining" and "planning" and as *specific* as "ontology alignment" and "information extraction". Such discrepancy among given queries poses particular challenges for accurate expert identification.

Previous studies usually formulate the problem of expert finding as a document search problem in the information retrieval community. Although promising results are obtained [37] by standard document search algorithms, the returned results are documents, not candidates. We take a social website as an example. Users actively participate in various online activities, such as posting, commenting, tagging, rating, and reviewing. The online textual information provides evidence for users' skills and expertise. Moreover, users engage in online communities, collaborating, and exchanging information with each other. Each user cannot be simply represented by her posts or comments and she has much more complicated personal, social, and

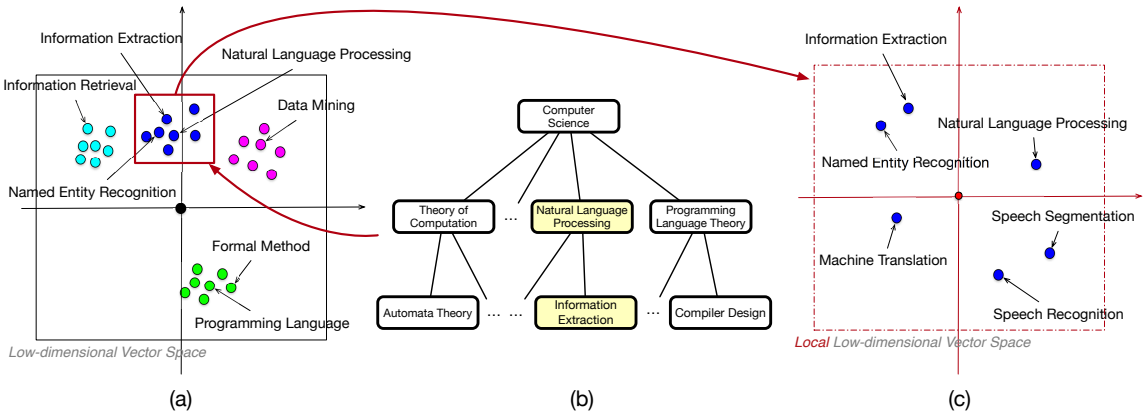


Figure 1.3: A toy example of locally-trained embedding learning with a concept hierarchy as guidance.

collaborative practices [25].

Many approaches have been proposed and studied for expert finding. The most popular models are the document-based generative probabilistic models [2, 3, 29]. The major idea of the document-based models is that the expertise of a candidate can be estimated by aggregating textual evidence from relevant documents, which is retrieved by statistical language models. Nevertheless, this method suffers from the following two drawbacks. On one hand, when applying the statistical language model, there is a vocabulary gap between terms in the query and the documents; on the other hand, such a method ignores network structure; that is the relationships among the candidates and other objects in the heterogeneous information network.

We attempt to solve the problem of expert finding, particularly focusing on specific queries with narrow semantic meanings without downgrading the accuracy for general ones. A novel framework based on query expansion is proposed. It includes two different components, one is textual analysis, to provide evidence for expertise identification, and the other is authority ranking, to rank the candidates in the heterogeneous bibliographical networks.

Locally-trained Embedding Learning via Concept Hierarchy.

In order to address the vocabulary gap, representations [13, 68, 101] learning is proposed to project the terms into a latent semantic space, such that terms with similar semantic meanings are close to each other in the latent vector space. The vector representations are also known as embeddings or distributed representations. The learned embeddings are based on the co-occurrence statistics derived from the whole corpus, which can be (loosely) interpreted as a low-rank approximation for the observation data in the corpus [13, 24, 58].

Nevertheless, information regarding some specific queries might be missing through the semantic matching method. We have a toy example shown in Figure 1.3(a), where terms related to different domains

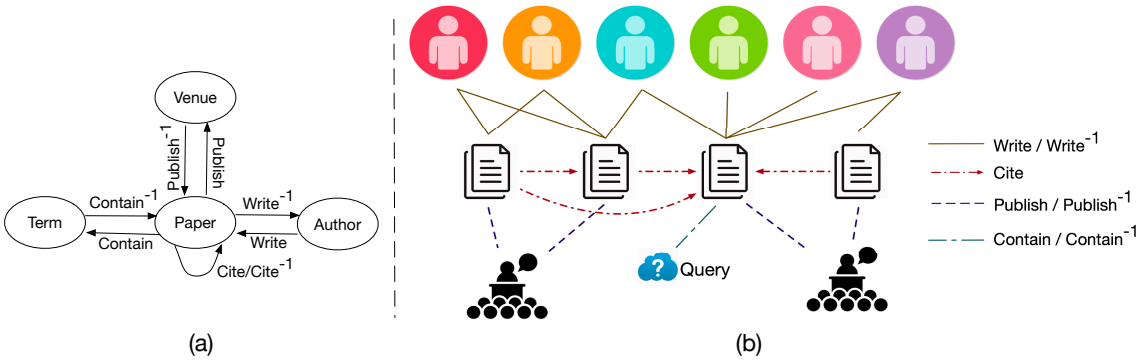


Figure 1.4: An example heterogeneous bibliographical network with four types of objects: authors, papers, venues, and terms. (a) the network schema; (b) a subnetwork where all the documents are relevant to the given query.

form different clusters, such as "information retrieval", "natural language processing", "data mining", and "programming language". Meanwhile, "information extraction" is close to both "natural language processing" and "named entity recognition". Particularly for the task of expert finding, if we expand the query "information extraction" to "natural language processing", there will be semantic drift.

In order to address the semantic drift discussed above, we propose to train a local embedding with concept hierarchy as guidance, as shown in Figure 1.3(b). For the query "information extraction", the *cluster* that "information extraction" belongs to can be identified as "natural language processing". Then the local embeddings can be learned based on the documents that are relevant to "natural language processing", as shown in Figure 1.3(c). Since the locally-trained embeddings only need to preserve the information respecting the cluster of "natural language processing", it has stronger representation power. Consequently, the local embeddings better capture the subtle semantic information such that "information extraction" shares closer semantic meaning with "named entity recognition", compared with "natural language processing".

Ranking within Relevance Network.

Extensive online textual information is available from candidates' activities, which serves as evidence for expertise identification. However, the final target of expert finding is to rank *candidates*, not textual information. There is a disparity.

The document-based models aggregate the relevant documents associated with each candidate and rank the candidates accordingly. The importance of each document is approximated by a monotonic function of the number of citations, such as logarithm functions. Such an aggregation method is inaccurate and sensitive to the choice of the monotonic function. On the other hand, besides textual information, the interactions among candidates and other objects (e.g., other candidates, group discussion in online social communities, venues in academia) offer additional insights for estimating the users' cognitive capabilities.

The interactions among the objects of different types naturally form a heterogeneous information network [89, 91]. Bibliographical information network is a typical heterogeneous information network, which characterizes the academic publication behaviors of researchers. In heterogeneous bibliographical networks, researchers have various activities, including publishing, collaborating, and attending venues. In Figure 1.4(a), the network schema of an example heterogeneous bibliographical network is depicted, with an illustration in Figure 1.4(b).

To close the gap between textual information analysis and candidate ranking, we propose a coupled random walk algorithm, including both inter-type random walks and intra-type random walks, to estimate the authority of objects in the network and the rank order of candidates. More concretely, the ranking algorithms considers the relative importance of different edge types in the heterogeneous bibliographical network.

In Section 6, we present the framework for the problem of expert finding in heterogeneous information networks. The new framework is called **LE-expert**, which is short for **L**ocally-trained **E**mbedding for **Expert** Finding. The proposed framework includes two phases; the first is locally-trained embedding learning with concept hierarchy as guidance, based on which we obtain query expansion for the given query; the second is the authority rank algorithm within the heterogeneous bibliographical network, which is retrieved and constructed based on the query expansion. Such a framework is particularly designed for specific queries.

1.4 Notation

We use lowercase letters $(a, b, \ldots)$ to denote scalars, bold lower case letters $(\mathbf{a}, \mathbf{b}, \ldots)$ for vectors, and bold upper case letters $(\mathbf{A}, \mathbf{B}, \ldots)$ for matrices, and high-order tensors by calligraphic upper case letters $(\mathcal{A}, \mathcal{B}, \ldots)$. A tensor is a higher order generalization of a vector (first order tensor) and a matrix (second order tensor). From a multi-linear algebra view, tensor is a multi-linear mapping over a set of vector spaces. For a real number a, we denote by $\lfloor a \rfloor$ the largest integer that is no greater than a. For a vector $\mathbf{x}$, define vector norm as $\|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2}$. Considering matrix $\mathbf{A}$, we denote by $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ the largest and smallest eigenvalue of $\mathbf{A}$, respectively. For a pair of matrices $\mathbf{A}, \mathbf{B}$ with commensurate dimensions, $\langle \mathbf{A}, \mathbf{B} \rangle$ denotes the trace inner product on matrix space that $\langle \mathbf{A}, \mathbf{B} \rangle := \text{trace}(\mathbf{A}^\top \mathbf{B})$. Given a matrix $\mathbf{A} \in \mathbb{R}^{m_1 \times m_2}$, its (ordered) singular values are denoted by $\gamma_1(\mathbf{A}) \geq \gamma_2(\mathbf{A}) \geq \cdots \geq \gamma_m(\mathbf{A}) \geq 0$ where $m = \min\{m_1, m_2\}$. Moreover, $M = \max\{m_1, m_2\}$. We also define $\|\cdot\|$ for various norms defined on matrices, based on the singular values, including nuclear norm $\|\mathbf{A}\|_* = \sum_{i=1}^m \gamma_i(\mathbf{A})$, spectral norm $\|\mathbf{A}\|_2 = \gamma_1(\mathbf{A})$, and the Frobenius norm $\|\mathbf{A}\|_F = \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle} = \sqrt{\sum_{i=1}^m \gamma_i^2(\mathbf{A})}$. In addition, we define $\|\mathbf{A}\|_\infty = \max_{1 \leq j \leq m_1, 1 \leq k \leq m_2} A_{jk}$, where A_{jk}

is the element of $\mathbf{A}$ at row j, column k. The order of tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times \ldots \times n_2 \times \ldots \times n_K}$ is K, where n_k is the dimensionality of the k-th order. Elements of $\mathcal{A}$ are denoted as $\mathcal{A}_{i_1 \ldots i_k \ldots i_n}, 1 \leq i_k \leq n_k$. We denote the number of elements in $\mathcal{A}$ by $N = \prod_{k=1}^{K} n_k$.

1.5 Organization of the Thesis

The rest of this thesis is organized as follows. In Chapter 2, the literature review is presented. In Chapter 3, a low-rank decomposition model with convex regularization is proposed for tensor decomposition with gross corruption, in the regime of high-dimensional statistical analysis. In Chapter 4, a unified low-rank estimation model with nonconvex penalty is proposed, including analysis for both matrix completion and matrix sensing, also in the regime of high-dimensional statistical analysis. In Chapter 5, embedding learning for high-dimensional heterogeneous event data is investigated, where there are strongly typed objects interacting in each event, and the proposed framework is based on hyper-edge modeling. In Chapter 6, regarding the problem of expert finding, locally-trained embedding learning with concept hierarchy as guidance is proposed to preserve the semantic information regarding specific queries with narrow semantic meanings. Chapter 7 concludes this thesis and points out some future directions.

Chapter 2

Literature Review

2.1 Low-rank Estimation Models

In this section, existing studies on low-rank estimation models are discussed. Two different categories are included, one is convex regularization and the other is nonconvex regularization.

2.1.1 Low-rank Estimation Models with Convex Regularization

Nuclear Norm. The problem of recovering an unknown (nearly) low-rank from limited information has received increasing interest due to its broad applications, and in particular the famous Netflix problem. Since it is not tractable to minimize the rank of a matrix directly, many surrogate loss functions of the matrix rank have been proposed (*e.g.*, nuclear norm [86, 19, 79, 71, 53], Schatten-p norm [82, 74], max norm [87, 15], the von Neumann entropy [52]). Among those surrogate losses for rank, nuclear norm is probably the most widely used penalty for low-rank matrix estimation [71, 53], since it is the tightest convex relaxation of the matrix rank. In [18], Candès and Recht show that most of the underlying low-rank matrices can be exactly recovered by sovling a convex problem. Their results can be stated as follows. Suppose the underlying matrix is $\mathbf{M} \in \mathbb{R}^{m \times m}$ and there are n sampled entries from $\mathbf{M}$ that $\{\mathbf{M}_{ij} : (i,j) \in \Omega\}$ where Ω is the set of cardinality n, then most matrices $\mathbf{M}$ of rank r can be exactly recovered by solving the minimization problem

$$\begin{aligned} &\text{minimization} \quad \|\mathbf{\Theta}\|_* \\ &\text{subject to} \quad \mathbf{\Theta}_{ij} = \mathbf{M}_{ij}, (i,j) \in \Omega. \end{aligned} \tag{2.1.1}$$

Where $\|\mathbf{\Theta}\|_*$ is the nuclear norm of $\mathbf{\Theta}$, also kown as nuclear norm, is defined as the sum of singular values of $\mathbf{\Theta}$. [14] proposes to solve (2.1.1) via a singular value thresholding algorithm. Nuclear norm regularization has been applied to various applications, including recommender system [18], image compression [79], dimension reduction in multivariate linear regression [95], computer vision [112].

It worth noting that there are many other surrogate loss functions of the matrix rank proposed besides

nuclear norm [86, 19, 79, 71, 53]. Other surrogate loss functions of the matrix rank including Schatten-p norm [82, 74], max norm [87, 15], the von Neumann entropy [52]). However, nuclear norm is probably the most widely used penalty for low-rank matrix estimation [71, 53], since it is the tightest convex relaxation of the matrix rank.

Gross Corruption. The problem of recovering the data under gross error has gained many attentions recently in matrix decomposition. A large body of work have been proposed and analyzed statistically. For example, [20] considered the problem of recovering an unknown low-rank and an unknown sparse matrix, given the sum of the two matrices. [16] proposed a similar problem, namely robust principal component analysis (RPCA), which studies the problem of recovering the low-rank and sparse matrices by solving a convex program. [32] studied multi-task regression which decomposes the coefficient matrix into two matrices, and imposes different group sparse regularization on two matrices. [115] considered more general case, where the parameter matrix could be the superposition of more than two matrices with different structurally constraints. This chapter extends [16] from two perspective: we extend the problem from matrices to high-order tensors, and consider the additional noise setting. We notice that [70] extended RPCA to tensors, which aims to recover the low-rank and sparse tensors by solving a constrained convex program. However, our formulation departs from [70] in that we consider not only the sparse corruption, but also the dense noise. We also note that low-rank noisy matrix completion [71] and robust matrix decomposition [1] [38] have been studied in the high dimensional setting as well. Our model can be seen as the high-order extension of robust matrix decomposition. This extension is nontrivial, because the treatment of the tensor nuclear norm (Schatten-1 norm) is more complicated. More importantly, for the robust matrix decomposition problem considered [1], only the sum of error bound of two matrices (low-rank matrix and the sparse corruption matrix) can be obtained under the assumption of restricted strongly convexity. In contrast, under a different condition, our analysis provides error bound for each tensor component (low-rank tensor and the sparse corruption tensor) separately, making our results more appealing in practice and of independent theoretical interest. Since the problem in [1] is a special case of our problem, our technical tool can be directly applied to their problem and yields new error bounds on the low-rank matrix as well as the sparse corruption matrix separately.

2.1.2 Low-rank Estimation Models with Nonconvex Penalty

Nonconvex penalty for matrix completion has been studied in various applications. In [74], the joint Schatten p-norm and ℓ_p norm are used to impose the low-rank structure, which is shown to be robust to outliers and

have better performance for real world applications, such as collaborative filter and link prediction in social networks. [39] proposes to use truncated nuclear norm to approximate the rank of a matrix. It is observed to have better empirical performance for both synthetic and real visual datasets. For problem of robust principal component analysis, [108] propose to recover the underlying low-rank matrice and sparse matrix via a nonconvex loss function and nonconvex penalty on the singular values. Empirically, the proposed estimator gives better accuracy in recovering the matrix. Similar studies are discussed in [61, 66, 117]. However, the theoretical justification for the nonconvex surrogates of matrix rank is still an open problem.

2.2 Embedding Learning Models

Literature regarding embedding learning in real world applications is presented. There are two lines: one line is about embedding learning in heterogeneous information netowrk and the other line is on expert finding.

2.2.1 Embedding Learning in Heterogeneous Information Networks

Heterogeneous information networks ubiquitously exist in real word and have been investigated in previous studies [45, 89]. Quite many methods were developed towards various applications including classification [45], clustering [89], and similarity search [106]. Recall that when the number of object types in each event is one, the heterogeneous information networks reduce to homogeneous. However, in previous studies of heterogeneous information networks, only binary interactions are studied. In this paper, we model the semantic relatedness among objects based on the concept of events, in which multiple binary interactions happen simultaneously. Via modeling heterogeneous information networks based on events, more subtle information can be captured.

In particular for the embedding task, both [92] and [21] study the problem of object embedding in heterogeneous information networks. But instead of modeling proximity among objects in each event as a whole, [92, 21] decompose each event into several binary interactions and then do the pairwise modeling separately. [59] studies the problem of a new angle by considering the multi-faceted representation of objects in information networks. [122] learns object embedding by considering the hetergenity of both nodes and relations, based on knowledge base, which is specifically developed for the task of recommender system. Our model is substantially different since we directly model each hyperedge as a whole so that the proximity among objects can be better preserved. [22] studies the problem of embedding in heterogeneous information network, specifically for the task of anomaly detection, with each event defined as a collection of categorical values. Our framework is more general and consider two different methods of modeling the proximity among

objects. [119] shares similar flavor as our modeling for task of modeling people's activities in urban space. These two papers are tailed for particular tasks; while our framework is general, which is can be adopted for various tasks via feeding corresponding task labels.

The hyperedge-based framework is also related to tensor analysis, with each event corresponding to an element int the tensor. Such studies in higher-order data [51, 47] have recently emerged for some tasks, such as recommender system [81], multi-relational learning [44], and clustering [5]. In [81], a tensor factorization model is designed specifically for tag recommendation; while we explore a more general framework for embedding from which two methods are designed to model the object-driven and hyperedge-driven proximity respectively. [5] defines higher-order network structures, such as cycles and feed-forward loops, and uses tensor to model the heterogenous event data. In sharp contrast, most of these methods cannot scale to the datasets used in this paper and meanwhile our framework is more general in the sense that it allows multiple event types. In addition, [5] only models the events with one type of object; while HEBE supports multiple object types in multiple event types. However, in order to perform tensor decomposition, the tensor needs to be materalized. Due to curse of dimension, such a method is not computationally feasible. In our case, each time we sample an event, which is independent of the size of dimensions of the corresponding tensor. Moreover, we adopted ASGD, which is designed for distributed computation, leading to better scability.

In addition, some dimension reduction methods can be adapted for object embedding learning in heterogeneous information networks, such as principal component analysis [105], singular value decomposition [105], and non-negative matrix factorization [57]. However, these methods ignores the intrinsic event types and fails to model the participating objects collectively, and thus cannot capture the intricate proximity in heterogeneous information networks.

2.2.2 Embedding Learning for Expert Finding

In general, there are two major approaches [3] for the probelm of expert finding, one is profile-based [64] and the other is document-based [2, 29] (also known as the candidate and document models). For the profile-based models, each candidate is represented via a set of terms. Given a query, the candidates are ranked via the ad-hoc retrieval models. In contrast, the document-based models are to firstly retrieve all the relevant documents of the query and then the candidates are ranked via aggregating the associated documents. Since the document-based models make use of the whole corpus, it is usually more effective compared with the profile-based ones [3, 26]. Besides those two models, there are many other approaches via taking advantage of additional information. For instance, Karimzadehgan et al. propose to solve the problem of expert finding via incorporating the organizational hierarchy [49]. The problem of vocabulary gap

is addressed via query expansion with Normalized Google Distance [116]. More recently, an unsupervised embedding learning method is proposed, where the embeddings are learned based on the co-occurrence between candidates and terms [101]. However, these methods mainly focus on the textual information with the rich network structure being ignored.

Regarding the (heterogeneous) network structure, it is proposed to rank the candidates within an online forum via a propagation-based approach [123]. Besides, the problem is formalized as searching for reliable users and content for the task of community-based query answering in a co-training fashion [8]. For the task of collaborative tagging recommendation, Noll et al. assess the expertise of users using a graph-based ranking method similar to the HITS algorithm [75]. Deng et al. propose a joint optimization framework to rank candidates based on the consistency implied by the network structureIn [25]. Moreover, there are some other relevant studies, such as co-rank [124] where authors and their publications are ranked based on a coupled random walk algorithm; NetClus [90] simultaneously ranks and clusters the strongly-typed objects in the heterogeneous information network; and RankClus [45] applies similar philosophy to classification and ranking. Nevertheless, these works are either query independent or consider the query-document relatedness based on global semantic mapping, which loses information for specific queries. Our method not only considers the network structure, but also captures the query expansion for specific queries based on locally-trained embeddings.

The idea of query expansion regarding local document analysis has been previously studied for information retrieval [114]. Global analysis and local feedback are combined for query expansion with a new weight ranking function for query expansion. Recently, Diaz et al. propose to do query expansion based on locally-trained embedding learning, to obtain query expansion for terms with ambiguous semantic meanings [27].

Chapter 3

Low-rank Decomposition Models with Convex Regularization

3.1 Notation and Background for Tensors

Before proceeding, we add more definitions on the notation and state assumptions that will appear in the following analysis. For more details about tensor algebra, please refer to [56].

The mode-k vectors of a K order tensor $\mathcal{A}$ are the n_k dimensional vectors obtained from $\mathcal{A}$ by varying index i_k while keeping the other indices fixed. The mode-k vectors are the column vectors of mode-k flattening matrix $\mathbf{A}_{(k)} \in \mathbb{R}^{n_k \times (n_1 \dots n_{k-1} n_{k+1} \dots n_K)}$ that results by mode-k flattening the tensor $\mathcal{A}$. For example, matrix column vectors are referred to as mode-1 vectors and matrix row vectors are referred to as mode-2 vectors.

The scalar product of two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n_1 \dots n_2 \dots n_K}$, is defined as $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1} \dots \sum_{i_K} \mathcal{A}_{i_1 \dots i_K} \mathcal{B}_{i_1 \dots i_K} = \text{vec}(\mathcal{A})\text{vec}(\mathcal{B})$, where $\text{vec}(\cdot)$ is a vectorization. The Frobenius norm of a tensor $\mathcal{A}$ is $|||\mathcal{A}|||_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$.

There are multiple ways to define tensor rank. In this chapter, following [98], we define the rank of a tensor based on the mode-k rank of a tensor. More specifically, the mode-k rank of a tensor $\mathcal{X}$, denoted by $\text{rank}_k(X)$, is the rank of the mode-k unfolding $\mathbf{X}_{(k)}$ (note that $\mathbf{X}_{(k)}$ is a matrix, so its rank is well-defined). Based on mode-k rank, we define the rank of tensor $\mathcal{X}$ as $r(\mathcal{X}) = (r_1, \dots, r_k)$ if the mode-k rank is r_k for $k = 1, \dots, K$. Note that the mode-k rank can be computed in polynomial time, because it boils down to computing a matrix rank, whereas computing tensor rank [56] is NP complete.

Similarly, we extend the nuclear norm (a.k.a. nuclear norm) of matrices [88] to tensors. The overlapped Schatten-1 norm is defined as $|||\mathcal{X}|||_{S_1} = \frac{1}{K} \sum_{k=1}^{K} \|\mathbf{X}_{(k)}\|_{S_1}$, where $\mathbf{X}_{(k)}$ is the mode-k unfolding of $\mathcal{X}$, and $\|\cdot\|_{S1}$ is the Schatten-1 norm for a matrix, $\|\mathbf{X}\|_{S_1} = \sum_{j=1}^{r} \sigma_j(\mathbf{X})$, where $\sigma_j(\mathbf{X})$ is the j-th largest singular value of $\mathbf{X}$. The dual norm of the Schatten-1 norm is Schatten-∞ norm (a.k.a., spectral norm) as $\|\mathbf{X}\|_{S_\infty} = \max_{j=1,\dots,r} \sigma_j(\mathbf{X})$.

By Hölder's inequality, we have $|\langle \mathbf{W}, \mathbf{X} \rangle| \leq \|\mathbf{W}\|_{S_1} \|\mathbf{X}\|_{S_\infty}$. It is easy to prove a similar result for the

overlapped Schatten-1 norm and its dual norm. We have the following Hölder-like inequality [98]:

$$|\langle \mathcal{W}, \mathcal{X} \rangle| \leq |||\mathcal{W}|||_{S_1} \, |||\mathcal{X}|||_{S_1^*} \leq |||\mathcal{W}|||_{S_1} \, |||\mathcal{X}|||_{\text{mean}} \,, \tag{3.1.1}$$

where $|||\mathcal{X}|||_{\text{mean}} := \frac{1}{K} \sum_{k=1}^{K} \|\mathbf{X}_{(k)}\|_{S_\infty}$.

Moreover, we define ℓ_1-norm and ℓ_∞-norm for tensors that $|||\mathcal{X}|||_1 = \sum_{i_1=1}^{n_1} \dots \sum_{i_K=1}^{n_K} |\mathcal{X}_{i_1,\dots,i_K}|$, $|||\mathcal{X}|||_\infty = \max_{1 \leq i_1 \leq n_1} \dots \max_{1 \leq i_K \leq n_K} |\mathcal{X}_{i_1,\dots,i_K}|$. By Hölder's inequality, we have $|\langle \mathcal{W}, \mathcal{X} \rangle| \leq |||\mathcal{W}|||_1 \, |||\mathcal{X}|||_\infty$, and the following inequality relates the overlapped Schatten-1 norm with the Frobenius norm,

$$|||\mathcal{X}|||_{S_1} \leq \frac{1}{K} \sum_{k=1}^{K} \sqrt{r_k} \, |||\mathcal{X}|||_F \,. \tag{3.1.2}$$

Let $\mathcal{W}^* \in \mathbb{R}^{n_1 \times \dots \times n_K}$ be the low-rank tensor that we wish to recover. We assume that $\mathcal{W}^*$ is of rank $(r_1, \dots, r_K)$. Thus, for each k, we have $\mathbf{W}^*_{(k)} = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^\top$, where $\mathbf{U}_k \in \mathbb{R}^{n_k \times r_k}$ and $\mathbf{V}_k \in \mathbb{R}^{r_k \times n_k}$ are orthogonal matrices, which consist of left and right singular vectors of $\mathbf{W}^*_{(k)}$, $\mathbf{S}_k \in \mathbb{R}^{r_k \times r_k}$ is a diagonal matrix whose diagonal elements are singular values. Let $\Delta \in \mathbb{R}^{n_1 \times \dots \times n_K}$ be an arbitrary tensor, we define the mode-k orthogonal complement $\mathbf{\Delta}''_k$ of its mode-k unfolding $\mathbf{\Delta}_{(k)} \in \mathbb{R}^{n_k \times \boldsymbol{N}}$ with respect to the true low-rank tensor $\mathcal{W}^*$ as follows

$$\mathbf{\Delta}''_k = (\mathbf{I}_{n_k} - \mathbf{U}_k \mathbf{U}_k^\top) \mathbf{\Delta}_{(k)} (\mathbf{I}_{\boldsymbol{N}} - \mathbf{V}_k \mathbf{V}_k^\top). \tag{3.1.3}$$

In addition $\mathbf{\Delta}'_k = \mathbf{\Delta}_{(k)} - \mathbf{\Delta}''_k$ is the component which has overlapped row/column space with the unfolding of the true tensor $\mathbf{W}^*_{(k)}$. Note that the decomposition $\mathbf{\Delta}_{(k)} = \mathbf{\Delta}'_k + \mathbf{\Delta}''_k$ is defined for each mode.

In [73], the concept of decomposibility and a large class of decomposable norms are discussed at length. Of particular relevance to us is the decomposability of the Schatten-1 norm and ℓ_1-norm. We have the following equality, i.e., mode-k decomposibility of the Schatten-1 norm that $\|\mathbf{W}^*_{(k)} + \mathbf{\Delta}''_k\|_{S_1} = \|\mathbf{W}^*_{(k)}\|_{S_1} + \|\mathbf{\Delta}''_k\|_{S_1}, k = 1, \dots, K$. To note that the decomposibility is defined on each mode. It is also easy to check the decomposibility of the ℓ_1-norm.

Let $\mathcal{V}^* \in \mathbb{R}^{n_1 \times \dots \times n_K}$ be the gross corruption tensor that we wish to recover. We assume the gross corruption is sparse, in that the cardinality $s = |\text{supp}(\mathcal{V}^*)|$ of its support, $S = \text{supp}(\mathcal{V}^*) = \big\{(i_1, i_2, \dots, i_K) \in [n_1] \times \dots \times [n_K] | \mathcal{V}^*_{i_1,\dots,i_K} \neq 0\big\}$. This assumption leads to the inequality between the ℓ_1 norm and the Forbenius norm that $|||\mathcal{V}^*|||_1 \leq \sqrt{s} \, |||\mathcal{V}^*|||_F$. Moreover, we have $|||\mathcal{V}^*|||_1 = |||\mathcal{V}^*_S|||_1$. For any $\mathcal{D} \in \mathbb{R}^{n_1 \times \dots \times n_K}$, we have $|||\mathcal{D}|||_1 = |||\mathcal{D}_S|||_1 + |||\mathcal{D}_{S^c}|||_1$.

3.2 Main Results

To get a deep theoretical insight into the recovery property of robust tensor decomposition, we will now present a set of estimation error bounds. Unlike the analysis in [1], where only the summation of the estimation errors on the low-rank matrix and gross corruption matrix are analyzed, we aim at obtaining the estimation error bounds on each tensor (the low-rank tensor and corrupted tensor) separately. All the proofs can be found in the appendix.

Instead of considering the observation model in 1.2.3, we consider the following more general observation model

$$y_i = \langle \mathcal{W}^*, \mathcal{X}_i \rangle + \langle \mathcal{V}^*, \mathcal{X}_i \rangle + \epsilon_i, i = 1, \ldots, M, \tag{3.2.1}$$

where $\mathcal{X}_i$ can be seen as an observation operator, and ϵ_i's are i.i.d. zero mean Gaussian noise with variance σ^2. Our goal is to estimate an unknown rank $(r_1, \ldots, r_k)$ of tensor $\mathcal{W}^* \in \mathbb{R}^{n_1 \times \ldots \times n_K}$, as well as the unknown support of tensor $\mathcal{V}^*$, from observations $y_i, i = 1, \ldots, M$. We propose the following convex minimization to estimate the unknown low-rank tensor $\mathcal{W}^*$ and the sparse corruption tensor $\mathcal{V}^*$ simultaneously, with composite regularizers on $\mathcal{W}$ and $\mathcal{V}$ as follows:

$$(\widehat{\mathcal{W}}, \widehat{\mathcal{V}}) = \arg\min_{\mathcal{W},\mathcal{V}} \frac{1}{2M} \|\mathbf{y} - \mathfrak{X}(\mathcal{W} + \mathcal{V})\|_2^2 + \lambda_M \left|\!\left|\!\left| \mathcal{W} \right|\!\right|\!\right|_{S_1} + \mu_M \left|\!\left|\!\left| \mathcal{V} \right|\!\right|\!\right|_1, \tag{3.2.2}$$

where $\mathbf{y} = (y_1, \ldots, y_M)^\top$ is the collection of observations, $\mathfrak{X}(\mathcal{W})$ is the linear observation model that $\mathfrak{X}(\mathcal{W}) = [\langle \mathcal{W}, \mathcal{X}_1 \rangle, \ldots, \langle \mathcal{W}, \mathcal{X}_M \rangle]^\top$. Note that (1.2.4) is a special case of (3.2.2), where the linear operator the identity tensor, we have y_i as observation of each element in the summation of tensors $\mathcal{W}^* + \mathcal{V}^*$.

We also define $\mathbf{y}^* = (y_1^*, \ldots, y_M^*)^\top$, where $y_i^* = \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i \rangle$, is the true evaluation. Due to the noise of observation model, we have $\mathbf{y} = \mathbf{y}^* + \boldsymbol{\epsilon}$. In addition, we define the adjoint operator of $\mathfrak{X}$ as $\mathfrak{X}^* : \mathbb{R}^M \to \mathbb{R}^{n_1 \times \ldots \times n_K}$ that $\mathfrak{X}^*(\boldsymbol{\epsilon}) = \sum_{i=1}^M \epsilon_i \mathcal{X}_i$.

3.2.1 Deterministic Bounds

This section is devoted to obtain the deterministic bound of the residual low-rank tensor $\Delta = \widehat{\mathcal{W}} - \mathcal{W}^*$ and residual corruption tensor $\mathcal{D} = \widehat{\mathcal{V}} - \mathcal{V}^*$ separately, which makes our analysis unique.

We begin with a key technical lemma on residual tensors $\Delta = \widehat{\mathcal{W}} - \mathcal{W}^*$ and $\mathcal{D} = \widehat{\mathcal{V}} - \mathcal{V}^*$, obtained from the convex problem in (3.2.2).

Lemma 3.2.1. *Let $\widehat{\mathcal{W}}$ and $\widehat{\mathcal{V}}$ be the solution of minimization problem (3.2.2) with $\lambda_M \geq 2 \left|\!\left|\!\left| \mathfrak{X}^*(\boldsymbol{\epsilon}) \right|\!\right|\!\right|_{\text{mean}} / M$,*

$\mu_M \geq 2\left\|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})\right|\right\|_\infty / M$, *we have*

1. *rank*$(\boldsymbol{\Delta}_k') \leq 2r_k$.

2. *There exist* $\beta_1 \geq 3$ *and* $\beta_2 \geq 3$, *such that* $\sum_{k=1}^K \|\boldsymbol{\Delta}_k''\|_{S_1} \leq \beta_1 \sum_{k=1}^K \|\boldsymbol{\Delta}_k'\|_{S_1}$ *and* $\left\|\left|\mathcal{D}_{S^c}\right|\right\|_1 \leq \boldsymbol{\beta}_2 \left\|\left|\mathcal{D}_S\right|\right\|_1$.

The lemma can be obtained by utilizing the optimality of $\widehat{\mathcal{W}}$ and $\widehat{\mathcal{V}}$, as well as the decomposibility of Schatten-1 norm and ℓ_1-norm of tensors.

Also, we obtain the key property of the optimal solution of (3.2.2), presented in the following theorem.

Theorem 3.2.2. *Let* $\widehat{\mathcal{W}}$ *and* $\widehat{\mathcal{V}}$ *be the solution of minimization problem* (3.2.2) *with* $\lambda_M \geq 2\left\|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})\right|\right\|_{\text{mean}} / M$, $\mu_M \geq 2\left\|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})\right|\right\|_\infty / M$, *we have*

$$\frac{1}{2M}\|\mathfrak{X}(\Delta + \mathcal{D})\|_2^2 \leq \frac{3\lambda_M}{2K} \sum_{k=1}^K \|\boldsymbol{\Delta}_k'\|_{S_1} + \frac{3\mu_M}{2} \left\|\left|\mathcal{D}_S\right|\right\|_1 . \tag{3.2.3}$$

Theorem 3.2.2 provides a deterministic prediction error bound for model (3.2.2). This is a very general result, and can be applied to any linear operator $\mathfrak{X}$, including the robust tensor decomposition case that we are particularly interested in this chapter. It also covers, for example, tensor regression, tensor compressive sensing, to mention a few.

Furthermore, we impose an assumption on the linear operator and the residual low-rank tensor and residue sparse corruption tensor, which generalized the restricted eigenvalue assumption [9] [32].

Assumption 3.2.3. *Defining* $\Omega = \{(\Delta, \mathcal{D}) | \sum_{k=1}^K \|\boldsymbol{\Delta}_k''\|_{S_1} \leq \beta_1 \sum_{k=1}^K \|\boldsymbol{\Delta}_k'\|_{S_1}, \left\|\left|\mathcal{D}_{S^c}\right|\right\|_1 \leq \beta_2 \left\|\left|\mathcal{D}_S\right|\right\|_1\}$, *we assume there exist positive scalars* κ_1, κ_2 *that*

$$\kappa_1 = \min_{\Delta, \mathcal{D} \in \Omega} \frac{\|\mathfrak{X}(\Delta + \mathcal{D})\|_2}{\sqrt{M}\left\|\left|\Delta\right|\right\|_F} > 0, \quad \kappa_2 = \min_{\Delta, \mathcal{D} \in \Omega} \frac{\|\mathfrak{X}(\Delta + \mathcal{D})\|_2}{\sqrt{M}\left\|\left|\mathcal{D}\right|\right\|_F} > 0.$$

Note that Assumption 3.2.3 is also related to restricted strong convexity assumption, which is proposed in [73] to analyze the statistical properties of general M-estimators in the high dimensional setting.

Combing the results in Theorem 3.2.2 and Assumption 3.2.3, we have the following theorem, which summarizes our main result.

Theorem 3.2.4. *Suppose Assumption 3.2.3 holds. Let* $\widehat{\mathcal{W}}, \widehat{\mathcal{V}}$ *be an optimal solution of* (3.2.2), *and take the*

regularization parameters $\lambda_M \geq 2\left|\left|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})\right|\right|\right|_{\text{mean}}/M$, $\mu_M \geq 2\left|\left|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})\right|\right|\right|_{\infty}/M$. *Then the following results hold:*

$$\left|\left|\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\right|\right|_F \leq \frac{3}{\kappa_1}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\lambda_M\sqrt{2r_k}}{\kappa_1} + \frac{\mu_M\sqrt{s}}{\kappa_2}\right), \tag{3.2.4}$$

$$\left|\left|\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\right|\right|_F \leq \frac{3}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\lambda_M\sqrt{2r_k}}{\kappa_1} + \frac{\mu_M\sqrt{s}}{\kappa_2}\right). \tag{3.2.5}$$

Theorem 3.2.4 provides us with the error bounds of each tensor separately. Specifically, these bounds not only measure how well our decomposition model can approximate the observation model defined in (4.1.1), but also measure how well the decomposition of the true low-rank tensor and gross corruption tensor is. When $s = 0$, our theoretical results reduce to that proposed in [98], which is a special case of our problem, i.e., noisy low-rank tensor decomposition without corruption.

On the other hand, the results obtained in Theorem 3.2.4 are very appealing both practically and theoretically. From the perspective of applications, this result is quite useful as it helps us to better understand the behavior of each tensor separately. From the theoretical point of view, this result is novel, and is incomparable with previous results [1][71] or simple generalization of previous results.

Though Theorem 3.2.4 has provided estimation error bounds of $\widehat{\mathcal{W}}$ and $\widehat{\mathcal{V}}$, it is unclear whether the rank of $\mathcal{W}^*$ and the support of $\mathcal{V}^*$ can be exactly recovered. We show that under some assumptions about the true tensors, both of them can be exactly recovered.

Corollary 3.2.5. *Under the same conditions of Theorem 3.2.4, if the following condition holds:*

$$\sigma_{r_k}(\mathbf{W}^*_{(k)}) > \frac{6(1+\beta_1)\sum_{k=1}^{K}\sqrt{2r_k}}{\kappa_1 MK}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\lambda_M\sqrt{2r_k}}{\kappa_1} + \frac{\mu_M\sqrt{s}}{\kappa_2}\right), \tag{3.2.6}$$

where $\sigma_{r_k}(\mathbf{W}^*_{(k)})$ *is the* r_k*-th largest singular value of* $\mathbf{W}^*_{(k)}$*, then*

$$\widehat{r}_k = \left\{\arg\max_r \sigma_r(\widehat{\mathbf{W}}_{(k)}) > \frac{3(1+\beta_1)\sum_{k=1}^{K}\sqrt{2r_k}}{\kappa_1 MK}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\lambda_M\sqrt{2r_k}}{\kappa_1} + \frac{\mu_M\sqrt{s}}{\kappa_2}\right)\right\}$$

recovers the rank of $\mathbf{W}^*_{(k)}$ *for all* k.

Furthermore, if the following condition holds:

$$\min_{i_1,\ldots,i_K}|\mathcal{V}^*_{i_1,\ldots,i_K}| > \frac{6(1+\beta_2)\sqrt{s}}{\kappa_2 M}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\lambda_M\sqrt{2r_k}}{\kappa_1} + \frac{\mu_M\sqrt{s}}{\kappa_2}\right), \tag{3.2.7}$$

then

$$\widehat{S} = \left\{ (i_1, i_2, \ldots, i_K) : \widehat{\mathcal{V}}_{i_1,\ldots,i_K} > \frac{3(1+\beta_2)\sqrt{s}}{\kappa_2 M} \left(\frac{1}{K} \sum_{k=1}^{K} \frac{\lambda_M \sqrt{2r_k}}{\kappa_1} + \frac{\mu_M \sqrt{s}}{\kappa_2} \right) \right\}$$

recovers the true support of $\mathcal{V}^*$.

Corollary 3.2.5, basically states that, under the assumption that the singular values of the low-rank tensor $\mathcal{W}^*$, and the entry values of corruption tensor $\mathcal{V}^*$ are above the noise level (e.g., (3.2.6) and (3.2.7)), we can recover the rank and the support successfully.

3.2.2 Noisy Tensor Decomposition

Now we are going back to study robust tensor decomposition with corruption in (1.2.4), which is a special case of (3.2.2), where the linear operator is identity tensor. As the linear operator $\mathfrak{X}$ is a vectorization such that $M = N$, and $\|\mathfrak{X}(\Delta + \mathcal{D})\|_2 = |||\Delta + \mathcal{D}|||_F$. In addition, it is easy to show that Assumption 3.2.3 holds with $\kappa_1 = \kappa_2 = O(1/\sqrt{N})$. It remains to bound $|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\text{mean}}$ and $|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\infty}$, as shown in the following lemma [1] [103].

Lemma 3.2.6. *Suppose that* $\mathfrak{X} : \mathbb{R}^{n_1 \times \cdots \times n_K} \to \mathbb{R}^N$ *is a vectorization of a tensor. Then we have with probability at least* $1 - 2\exp(-C(n_k + \boldsymbol{N})) - 1/N$ *that*

$$|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\text{mean}} \leq \frac{\sigma}{K} \sum_{k=1}^{K} \left(\sqrt{n_k} + \sqrt{\bar{N}_{\backslash k}} \right),$$
$$|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\infty} \leq 4\sigma\sqrt{\log N},$$

where C *is a universal constant.*

With Theorem 3.2.4 and Lemma 3.2.6, we immediately have the following estimation error bounds for robust tensor decomposition.

Theorem 3.2.7. *Suppose Assumption 3.2.3 holds. Then for the regularization constants*

$$\lambda_N \geq 2\sigma \sum_{k=1}^{K} \left(\sqrt{n_k} + \sqrt{\bar{N}_{\backslash k}} \right) / (NK),$$

$\mu_N > 8\sigma\sqrt{\log N}/N$, *with probability at least* $1 - 2\exp(-C(n_k + \bar{N}_{\backslash k})) - 1/N$, *any solution of* (1.2.4) *have*

the following error bound:

$$\left|\left|\left|\widehat{\mathcal{W}}-\mathcal{W}^*\right|\right|\right|_F \leq \frac{6}{\kappa_1}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\sigma\sum_{k=1}^{K}\left(\sqrt{n_k}+\sqrt{\bar{N}_{\backslash k}}\right)\sqrt{2r_k}}{\kappa_1 NK}+\frac{4\sigma\sqrt{s\log N}}{\kappa_2 N}\right),$$
$$\left|\left|\left|\widehat{\mathcal{V}}-\mathcal{V}^*\right|\right|\right|_F \leq \frac{6}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{\sigma\sum_{k=1}^{K}\left(\sqrt{n_k}+\sqrt{\bar{N}_{\backslash k}}\right)\sqrt{2r_k}}{\kappa_1 NK}+\frac{4\sigma\sqrt{s\log N}}{\kappa_2 N}\right).$$

In the special case that $n_1 = \ldots = n_K = n$ and $r_1 = \ldots = r_K = r$, we have $\left|\left|\left|\widehat{\mathcal{W}}-\mathcal{W}^*\right|\right|\right|_F = O\big(\sigma\sqrt{rn^{K-1}}+\sigma\sqrt{Ks\log n}\big)$ and $\left|\left|\left|\widehat{\mathcal{V}}-\mathcal{V}^*\right|\right|\right|_F = O\big(\sigma\sqrt{rn^{K-1}}+\sigma\sqrt{Ks\log n}\big)$, which matches the error bound of robust matrix decomposition [1] when $K = 2$.

Note that the high probability support and rank recovery guarantee for the special case of tensor decomposition follows immediately from Corollary 3.2.5. Due to the space limit, we omit the result here.

3.3 Algorithm

In this section, we present an algorithm to solve (1.2.4). Since (1.2.4) is a special case of (3.2.2), we consider the more general problem (3.2.2). It is easy to show that (3.2.2) is equivalent to the following problem with auxiliary variables Ψ, Φ:

$$\min_{\mathcal{W},\mathcal{V},\mathcal{Y},\mathcal{Z}} \frac{1}{2M}\|\mathbf{y}-\mathbf{x}^\top(\mathbf{w}+\mathbf{v})\|_2^2+\frac{\lambda_M}{K}\sum_{k=1}^{K}\left|\left|\left|\Psi_k\right|\right|\right|_{S_1}+\frac{\mu_M}{K}\sum_{k=1}^{K}\left|\left|\left|\Phi_k\right|\right|\right|_1,$$
$$\text{subject to } \mathbf{P}_k\mathbf{w}=\boldsymbol{\psi}_k, \mathbf{P}_k\mathbf{v}=\boldsymbol{\phi}_k,$$

where $\mathbf{x}, \mathbf{w}, \mathbf{v}, \boldsymbol{\psi}_k, \boldsymbol{\phi}_k$ are the vectorizations of $\sum_{i=1}^{M}\mathcal{X}_i, \mathcal{W}, \mathcal{V}, \Psi_k, \Phi_k$ respectively, and $\mathbf{P}_k$ is the transformation matrix that change the order of rows and columns so that $\mathbf{P}_k\mathbf{w} = \boldsymbol{\psi}_k$.

The augmented Lagrangian (AL) function of the above minimization problem with respect to the primal variables $(\mathcal{W}^t, \mathcal{V}^t)$ is given as follows:

$$\begin{aligned}
&L_\eta(\mathcal{W},\mathcal{V},\{\Psi_k\}_{k=1}^K,\{\Phi_k\}_{k=1}^K,\{\boldsymbol{\alpha}_k\}_{k=1}^K,\{\boldsymbol{\beta}_k\}_{k=1}^K)\\
=&\frac{1}{2}\|\mathbf{y}-\mathbf{x}^\top(\mathbf{w}+\mathbf{v})\|_2^2+\frac{\lambda_M M}{K}\sum_{k=1}^{K}\left|\left|\left|\Psi_k\right|\right|\right|_{S_1}+\frac{\mu_M M}{K}\sum_{k=1}^{K}\left|\left|\left|\Phi_k\right|\right|\right|_1\\
+&\eta\left(\sum_k(\boldsymbol{\alpha}_k^\top(\mathbf{P}_k\mathbf{w}-\boldsymbol{\psi}_k)+\frac{1}{2}\|\mathbf{P}_k\mathbf{w}-\boldsymbol{\psi}_k\|_2^2)+\sum_k(\boldsymbol{\beta}_k^\top(\mathbf{P}_k\mathbf{v}-\boldsymbol{\phi}_k)+\frac{1}{2}\|\mathbf{P}_k\mathbf{v}-\boldsymbol{\phi}_k\|_2^2)\right),
\end{aligned}$$

where $\boldsymbol{\alpha}^t, \boldsymbol{\beta}^t$ are Lagrangian multiplier vectors, and $\eta > 0$ is a penalty parameter.

We then apply the algorithm of Alternating Direction Method of Multipliers (ADMM) [10, 96] to solve the above optimization problem. Starting from initial points $(\mathbf{w}^0, \mathbf{v}^0, \{\Psi_k^0\}_{k=1}^K, \{\Phi_k^0\}_{k=1}^K, \{\boldsymbol{\alpha}_k^0\}_{k=1}^K, \{\boldsymbol{\beta}_k^0\}_{k=1}^K)$, ADMM performs the following updates iteratively:

$$
\begin{aligned}
\mathbf{w}^{t+1} &= \left((\mathbf{x}^\top \mathbf{y} - \mathbf{x}^\top \mathbf{x}\mathbf{v}^t) + \eta \sum_{k=1}^K \mathbf{P}_k^\top (\boldsymbol{\psi}_k^t - \boldsymbol{\alpha}_k^t) \right) / (1 + \eta K), \\
\mathbf{v}^{t+1} &= \left((\mathbf{x}^\top \mathbf{y} - \mathbf{x}^\top \mathbf{x}\mathbf{w}^{t+1}) + \eta \sum_{k=1}^K \mathbf{P}_k^\top (\boldsymbol{\phi}_k^t - \boldsymbol{\beta}_k^t) \right) / (1 + \eta K), \\
\Psi_k^{t+1} &= \mathrm{prox}^{tr}_{\frac{\lambda M}{\eta K}}(\mathbf{P}_k \mathbf{w}^{t+1} + \boldsymbol{\alpha}_k^t), \qquad \Phi_k^{t+1} = \mathrm{prox}^{\ell_1}_{\frac{\mu M}{\eta K}}(\mathbf{P}_k \mathbf{v}^{t+1} + \boldsymbol{\beta}_k^t) \qquad k = 1, \ldots, K, \\
\boldsymbol{\alpha}_k^{t+1} &= \boldsymbol{\alpha}_k^{t+1} + (\mathbf{P}_k \mathbf{w}^{t+1} - \boldsymbol{\psi}_k^{t+1}) \qquad \boldsymbol{\beta}_k^{t+1} = \boldsymbol{\beta}_k^{t+1} + (\mathbf{P}_k \mathbf{v}^{t+1} - \boldsymbol{\phi}_k^{t+1}) \qquad k = 1, \ldots, K,
\end{aligned}
$$

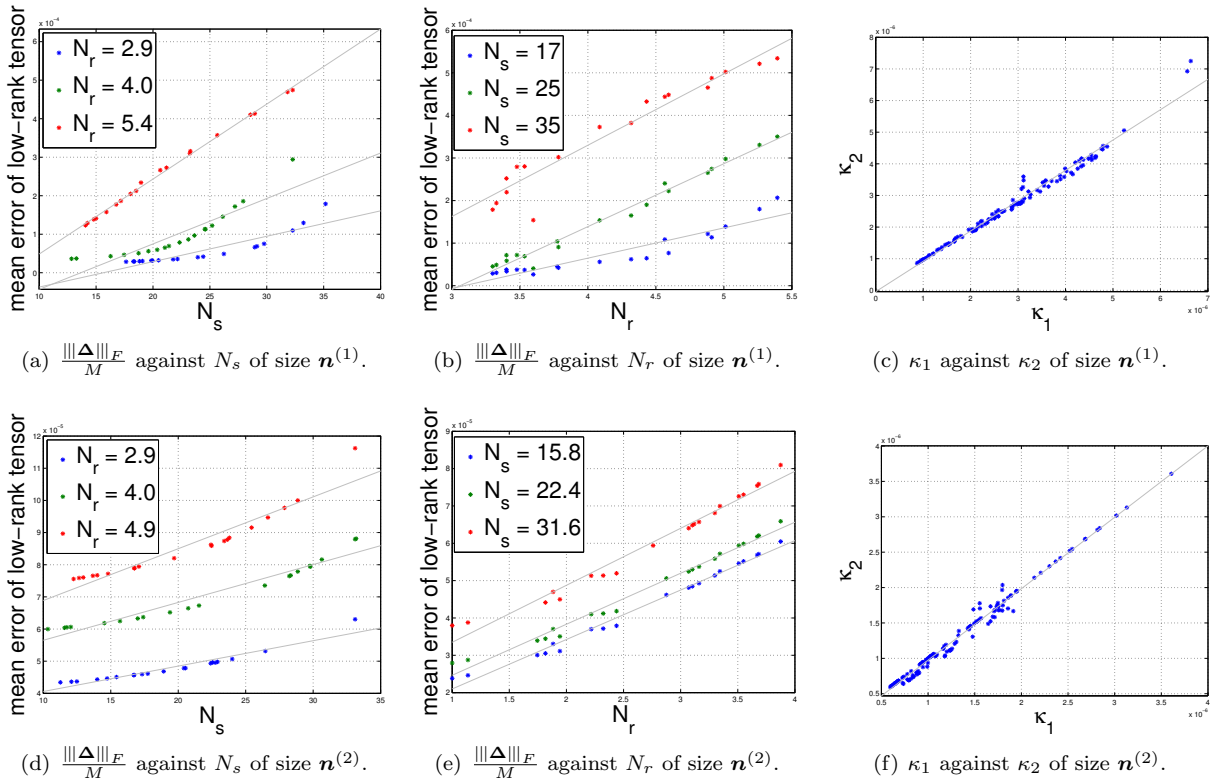
where $\mathrm{prox}^{tr}_\gamma(\cdot)$ is the soft-thresholding operator for nuclear norm, and $\mathrm{prox}^{\ell_1}_\gamma(\cdot)$ is the soft-thresholding operator for ℓ_1 norm [14, 35]. The stopping criterion is that all the partial (sub)gradients are (near) zero, under which condition we obtain the saddle point of the augmented Lagrangian function. Since (3.2.2) is strictly convex, the saddle point is the global optima for the primal problem.

3.4 Experiments

In this section, we conduct numerical experiments to confirm our analysis in previous sections. The experiments are conducted under the setting of robust noisy tensor decomposition.

We follow the procedure described in [98] for the experimental part. We randomly generate low-rank tensors of dimensions $\boldsymbol{n}^{(1)} = (50, 50, 20)$ (results are shown in Figure 3.1(a, b, c)) and $\boldsymbol{n}^{(2)} = (100, 100, 50)$(results are shown in Figure 3.1(d, e, f)) for various rank $(r_1, r_2, ..., r_k)$. Given a specific rank, we first generated the "core tensor" with elements $r_1 \times \ldots \times r_K$ from the standard normal distribution, and then multiplied each mode of the core tensor with an orthonormal factor randomly drawn from the Haar measure. For the gross corruption, we randomly generated the sparsity of the corruption matrix s, and then randomly selected s elements in which we put values randomly generated from uniform distribution. The additive independent Gaussian noise with variance σ^2 was added to the observations of elements. We use the alternating direction method of multipliers (ADMM) to solve the minimization problem (1.2.4). The whole experiments were repeated 50 times and the averaged results are reported.

The results are shown in Figure 3.1, where $N_r = \sum_{k=1}^K \sqrt{r_k}/K$, and $N_s = \sqrt{s}$. In Figure 3.1(a, d), we first fix N_r at different values, and then draw the value of $\left|\left|\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\right|\right|_F / N$ against N_s. Similarly, in Figure 3.1(b, e), we first fix N_s at different values, and then draw $\left|\left|\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\right|\right|_F / N$ against N_r. In Figure 3.1(c, f), we study



(a) $\frac{|||\mathbf{\Delta}|||_F}{M}$ against N_s of size $\boldsymbol{n}^{(1)}$. (b) $\frac{|||\mathbf{\Delta}|||_F}{M}$ against N_r of size $\boldsymbol{n}^{(1)}$. (c) κ_1 against κ_2 of size $\boldsymbol{n}^{(1)}$.

(d) $\frac{|||\mathbf{\Delta}|||_F}{M}$ against N_s of size $\boldsymbol{n}^{(2)}$. (e) $\frac{|||\mathbf{\Delta}|||_F}{M}$ against N_r of size $\boldsymbol{n}^{(2)}$. (f) κ_1 against κ_2 of size $\boldsymbol{n}^{(2)}$.

Figure 3.1: Results of robust noisy tensor decomposition with corruption, under different sizes.

the values of κ_1 and κ_2 at various settings. We can see that $\left|\left|\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\right|\right|_F / N$ scales linearly with both N_s and N_r. Similar scalings of $\left|\left|\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\right|\right|_F / N$ can be observed, hence we omit them due to space limitation. We can also observe from Figure 3.1(c, f) that, under various settings, $\kappa_1 \approx \kappa_2$, this finding is consistent with the fact that $\left|\left|\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\right|\right|_F / N \approx \left|\left|\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\right|\right|_F / N$. All these results are consistent with each other, validating our theoretical analysis.

Chapter 4

Low-Rank Estimation Models with Nonconvex Penalty

4.1 Low-rank Matrix Estimation with Nonconvex Penalty

In this section, we present a unified framework for low-rank matrix estimation with nonconvex penalty, followed by the theoretical analysis of the proposed estimator.

4.1.1 The Observation Model

We consider a generic observation model as follows:

$$y_i = \langle \mathbf{X}_i, \mathbf{\Theta}^* \rangle + \epsilon_i \quad \text{for } i = 1, 2, \ldots, n, \tag{4.1.1}$$

where $\{\mathbf{X}_i\}_{i=1}^n$ is a sequence of observation matrices, and $\{\epsilon_i\}_{i=1}^n$ are i.i.d. zero mean sub-Gaussian observation noise with variance σ^2. Moreover, the observation model can be rewritten in a more compact way as $\mathbf{y} = \mathfrak{X}(\mathbf{\Theta}^*) + \boldsymbol{\epsilon}$, where $\mathbf{y} = (y_1, \ldots, y_n)^\top$, $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$, and $\mathfrak{X}(\cdot)$ is a linear operator that $\mathfrak{X}(\mathbf{\Theta}^*) := (\langle \mathbf{X}_1, \mathbf{\Theta}^* \rangle, \langle \mathbf{X}_2, \mathbf{\Theta}^* \rangle, \cdots, \langle \mathbf{X}_n, \mathbf{\Theta}^* \rangle)^\top$. In addition, we define the adjoint of the operator $\mathfrak{X}$ as $\mathfrak{X}^* : \mathbb{R}^n \to \mathbb{R}^{m_1 \times m_2}$, which is defined as $\mathfrak{X}^*(\boldsymbol{\epsilon}) = \sum_{i=1}^n \epsilon_i \mathbf{X}_i$. It is worth noting that the observation model presented in (4.1.1), by which many matrix estimation problems can be unified, has also been considered before by [53, 71].

4.1.2 Examples

Low-rank matrix estimation has broad applications. We briefly review two examples: matrix completion and matrix sensing. For more examples, please refer to [53, 71].

Example 4.1.1 (Matrix Completion)**.** *In the setting of matrix completion with noise, one uniformly observes partial entries of the unknown matrix $\mathbf{\Theta}^*$ with noise. In detail, the observation matrix $\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2}$ is in the form of $\mathbf{X}_i = \mathbf{e}_{j_i}(m_1)\mathbf{e}_{k_i}(m_2)^\top$, where $\mathbf{e}_{j_i}(m_1)$ and $\mathbf{e}_{j_i}(m_2)$ are the canonical basis vectors in $\mathbb{R}^{m_1}$ and $\mathbb{R}^{m_2}$, respectively.*

Example 4.1.2 (Matrix Sensing)**.** *In the setting of matrix sensing, one observes a set of random projections of the unknown matrix* $\mathbf{\Theta}^*$*. More specifically, the observation matrix* $\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2}$ *has i.i.d. standard normal* $N(0,1)$ *entries, so that one makes observations of the form* $y_i = \langle \mathbf{X}_i, \mathbf{\Theta}^* \rangle + \epsilon_i$*. It is obvious that matrix sensing is an instance of the model* (4.1.1)*.*

4.1.3 The Proposed Estimator

We now propose an estimator that is naturally designed for estimating low-rank matrices. Given a collection of n samples $\mathcal{Z}_1^n = \{(y_i, \mathbf{X}_i)\}_{i=1}^n$, which is assumed to be generated from the observation model (4.1.1), the unknown low-rank matrix $\mathbf{\Theta}^* \in \mathbb{R}^{m_1 \times m_2}$ can be estimated by solving the following optimization problem

$$\widehat{\mathbf{\Theta}} = \underset{\mathbf{\Theta} \in \mathbb{R}^{m_1 \times m_2}}{\operatorname{argmin}} \frac{1}{2n} \big\| \boldsymbol{y} - \mathfrak{X}(\mathbf{\Theta}) \big\|_2^2 + \mathcal{P}_\lambda(\mathbf{\Theta}), \tag{4.1.2}$$

which includes two components: (i) the empirical loss function $\mathcal{L}_n(\mathbf{\Theta}) = (2n)^{-1} \|\boldsymbol{y} - \mathfrak{X}(\mathbf{\Theta})\|_2^2$; and (ii) the nonconvex penalty [28, 120, 121] $\mathcal{P}_\lambda(\mathbf{\Theta})$ with regularization parameter λ, which helps to enforce the low-rank structure constraint on the regularized M-estimator $\widehat{\mathbf{\Theta}}$. Considering the low rank assumption on the matrices, we apply the nonconvex regularization on the singular values of $\mathbf{\Theta}$, which induces sparsity of singular values, and therefore low-rankness of the matrix. For singular values of $\mathbf{\Theta}$, $\boldsymbol{\gamma}(\mathbf{\Theta}) = \big(\gamma_1(\mathbf{\Theta}), \gamma_2(\mathbf{\Theta}), \ldots, \gamma_m(\mathbf{\Theta})\big)$, where $\gamma_1(\mathbf{\Theta}) \geq \ldots \geq \gamma_m(\mathbf{\Theta}) \geq 0$, we define $\mathcal{P}_\lambda(\mathbf{\Theta}) = \sum_{i=1}^n p_\lambda\big(\gamma_i(\mathbf{\Theta})\big)$, where p_λ is a univariate nonconvex function. There is a line of research on nonconvex regularization and various nonconvex penalties have been proposed, such as SCAD [28] and MCP [120]. We take SCAD and MCP penalties as illustrations. Hence, for SCAD, the function $p_\lambda(\cdot)$ is defined as follows

$$p_\lambda(t) = \begin{cases} \lambda|t|, & \text{if } |t| \leq \lambda, \\ -\frac{t^2 - 2b\lambda|t| + \lambda^2}{2(b-1)}, & \text{if } \lambda < |t| \leq b\lambda, \\ (b+1)\lambda^2/2, & \text{if } |t| > b\lambda, \end{cases}$$

where $b > 2$ and $\lambda > 0$. The SCAD penalty corresponds to a quadratic spline function with knots at $t = \lambda$ and $t = b\lambda$. Regarding MCP, we have

$$\begin{aligned} p_\lambda(t) &= \lambda \int_0^{|t|} \Big(1 - \frac{z}{\lambda b}\Big)_+ dz \\ &= \Big(\lambda|t| - \frac{t^2}{2b}\Big) \mathbf{1}(|t| \leq b\lambda) + \frac{b\lambda^2}{2} \mathbf{1}(|t| > b\lambda), \end{aligned}$$

where $b > 0$ is a fix parameter.

Algorithm 1 $\{\boldsymbol{\Theta}^t\}_{t=1}^{K+1} \leftarrow \mathrm{PGH}(\lambda_0, \lambda_{\mathrm{tgt}}, \epsilon_{\mathrm{opt}}, L_{\min})$

1: **Input** $\lambda_0 > 0, \lambda_{\mathrm{tgt}} > 0, \epsilon_{\mathrm{opt}} > 0, L_{\min} > 0$
2: **parameters** $\eta \in (0,1), \delta \in (0,1)$
3: **initialize** $\boldsymbol{\Theta}^0 \leftarrow \mathbf{0}, L_0 \leftarrow L_{\min}, K \leftarrow \left\lfloor \frac{\ln(\lambda_0/\lambda_{tgt})}{\ln(1/\eta)} \right\rfloor$
4: **for** $t = 0, 1, 2, \ldots, K-1$ **do**
5: $\quad \lambda_{t+1} \leftarrow \eta\lambda_t$
6: $\quad \epsilon_{t+1} \leftarrow \lambda_t/4$
7: $\quad \{\boldsymbol{\Theta}^{t+1}, L_{t+1}\} \leftarrow \mathrm{ProxGrad}(\lambda_{t+1}, \epsilon_{t+1}, \boldsymbol{\Theta}^t, L_t)$
8: **end for**
9: $\{\boldsymbol{\Theta}^{K+1}, L_{K+1}\} \leftarrow \mathrm{ProxGrad}(\lambda_{\mathrm{tgt}}, \epsilon_{\mathrm{opt}}, \boldsymbol{\Theta}^K, L_K)$
10: **return** $\{\boldsymbol{\Theta}^t\}_{t=1}^{K+1}$

In addition, the nonconvex penalty $p_\lambda(t)$ can be further decomposed as $p_\lambda(t) = \lambda|t| + q_\lambda(t)$, where $|t|$ is the ℓ_1 penalty and $q_\lambda(t)$ is a concave component. For the SCAD penalty, $q_\lambda(t)$ can be obtained as follows,

$$q_\lambda(t) = -\big(|t| + \lambda\big)^2/\big(2(b-1)\big)\mathbf{1}(\lambda < |t| \le b\lambda) + \big(1/2(b+1)\lambda^2 - \lambda|t|\big)\mathbf{1}(|t| > b\lambda).$$

For MCP, the concave part is

$$q_\lambda(t) = -\frac{t^2}{2b}\mathbf{1}(|t| \le b\lambda) + \left(\frac{b\lambda^2}{2} - \lambda|t|\right)\mathbf{1}(|t| > b\lambda).$$

Since the regularization term $\mathcal{P}_\lambda(\boldsymbol{\Theta})$ is imposed on the vector of singular values, hence, the decomposability of $p_\lambda(t)$ is equivalent to the decomposability of $\mathcal{P}_\lambda(\boldsymbol{\Theta})$ as $\mathcal{P}_\lambda(\boldsymbol{\Theta}) = \lambda\|\boldsymbol{\Theta}\|_* + \mathcal{Q}_\lambda(\boldsymbol{\Theta})$, where $\mathcal{Q}_\lambda(\boldsymbol{\Theta})$ is the concave component, $\mathcal{Q}_\lambda(\boldsymbol{\Theta}) = \sum_{i=1}^m q_\lambda\big(\gamma_i(\boldsymbol{\Theta})\big)$, and $\|\boldsymbol{\Theta}\|_*$ is the nuclear norm.

4.1.4 Optimization Algorithm

In this section, we present a proximal gradient homotopy algorithm, which is adapted from [113], as shown in Algorithm 1, to solve the optimization problem with nonconvex penalty (4.1.2).

The main idea of proximal gradient homotopy method (PGH) is to solve the optimization problem with an initial regularization parameter $\lambda = \lambda_0$ that is sufficiently large and then gradually decrease λ until the target regularization parameter λ_{tgt} is attained, which will be given in Theorem 4.2.4 and Theorem 4.2.5, respecting different conditions.

In addition, we have $\lambda_t = \eta^t\lambda_0$, where η is an absolute constant. The number of iterations for the homotopy algorithm is $K = \lfloor \ln(\lambda_0/\lambda_{tgt})/\ln(1/\eta) \rfloor$. For the final stage of the proximal gradient homotopy method, we need to solve up to high precision with $\epsilon_{\mathrm{opt}} \ll \lambda_{\mathrm{tgt}}/4$. The key component in Algorithm 1 is the function ProxGrad() (Line 6 and 8), a proximal gradient method tailored for the M-estimator with nonconvex penalty, as shown in Algorithm 2. The details of the proximal gradient algorithm are introduced

as follows.

Recall that $\mathcal{P}_\lambda(\mathbf{\Theta}) = \lambda\|\mathbf{\Theta}\|_* + \mathcal{Q}_\lambda(\mathbf{\Theta})$. We define

$$\phi_\lambda(\mathbf{\Theta}) = \mathcal{L}_n(\mathbf{\Theta}) + \mathcal{P}_\lambda(\mathbf{\Theta}) = \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) + \lambda\|\mathbf{\Theta}\|_*, \tag{4.1.3}$$

where $\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) = \mathcal{L}_n(\mathbf{\Theta}) + \mathcal{Q}_\lambda(\mathbf{\Theta})$. For any fixed matrix $\mathbf{M}$ and a given regularization parameter λ, we define a local model of $\phi_\lambda(\mathbf{\Theta})$ around $\mathbf{M}$ using a simple quadratic approximation of $\widetilde{\mathcal{L}}_{n,\lambda}(\cdot)$ as follows:

$$\psi_{L,\lambda}(\mathbf{\Theta}; \mathbf{M}) = \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{M}) + \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{M})^\top(\mathbf{\Theta} - \mathbf{M}) + \frac{L}{2}\|\mathbf{\Theta} - \mathbf{M}\|_F^2 + \lambda\|\mathbf{\Theta}\|_*. \tag{4.1.4}$$

Suppose $\mathcal{T}_{L,\lambda}(\mathbf{M})$ is the unique minimize of $\psi_{L,\lambda}(\mathbf{\Theta}; \mathbf{M})$,

$$\mathcal{T}_{L,\lambda}(\mathbf{M}) = \operatorname*{argmin}_{\mathbf{\Theta}} \psi_{L,\lambda}(\mathbf{\Theta}; \mathbf{M}). \tag{4.1.5}$$

Via exploiting the structure of the nuclear norm regularization in (4.1.4), the optimization problem in (4.1.5) can be easily solved by singular value thresholding method [46, 14].

Suppose $\widehat{\mathbf{\Theta}}$ is the global solution to the optimization problem (4.1.2). According to the optimality condition, there exists $\mathbf{\Upsilon} \in \partial\|\widehat{\mathbf{\Theta}}\|_*$ such that, for all $\mathbf{\Theta} \in \mathbb{R}^{m_1 \times m_2}$,

$$(\widehat{\mathbf{\Theta}} - \mathbf{\Theta})^\top\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda\mathbf{\Upsilon}\big) \leq 0. \tag{4.1.6}$$

Hence, based on the optimality condition in (4.1.6), we measure the suboptimality of a $\mathbf{\Theta} \in \mathbb{R}^{m_1 \times m_2}$ using

$$\omega_\lambda(\mathbf{\Theta}) = \min_{\mathbf{\Upsilon}' \in \partial\|\widehat{\mathbf{\Theta}}\|_*} \max_{\mathbf{\Theta}'} \left\{ \frac{(\mathbf{\Theta} - \mathbf{\Theta}')^\top\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) + \lambda\mathbf{\Upsilon}'\big)}{\|\mathbf{\Theta} - \mathbf{\Theta}'\|_*} \right\} = \min_{\mathbf{\Upsilon}' \in \partial\|\widehat{\mathbf{\Theta}}\|_*} \Big\{ \big\|\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) + \lambda\mathbf{\Upsilon}'\big\|_2 \Big\},$$

where the second equality follows from the duality between $\|\cdot\|_*$ and $\|\cdot\|_2$. The main idea of the suboptimality is that, if $\mathbf{\Theta}$ is an exact optimum, by the optimality condition (4.1.6), we have $\omega_\lambda(\mathbf{\Theta}) < 0$; otherwise, if $\mathbf{\Theta}$ is close to the optimum, $\omega_\lambda(\mathbf{\Theta})$ is likely to be a small positive value.

To use Algorithm 2, we need to choose an initial optimistic estimate $L_{\min}$ for the Lipschitz constant $L_{\widetilde{\mathcal{L}}_{n,\lambda}}$, such that $0 < L_{\min} \leq L_{\widetilde{\mathcal{L}}_{n,\lambda}}$. The detailed discussion on Lipschitz constant $L_{\widetilde{\mathcal{L}}_{n,\lambda}}$ will be presented in Section 4.2.

Line 3 in Algorithm 2 is the line search algorithm (Algorithm 3), adaptively searching for the best quadratic coefficient L_k for the local quadratic approximation in (4.1.4).

Particularly, following the analysis in [113, 110], the iterative solution sequence $\big\{\mathbf{\Theta}^t\big\}_{t=1}^{K+1}$, which is

Algorithm 2 $\{\widetilde{\boldsymbol{\Theta}}, \widehat{L}\} \leftarrow \text{ProxGrad}(\lambda, \widehat{\epsilon}, \boldsymbol{\Theta}^0, L_0)$

Input $\lambda > 0, \widehat{\epsilon} > 0, \boldsymbol{\Theta}^0 \in \mathbb{R}^{m_1 \times m_2}, L_0 > 0, k = 0$
repeat
  $k \leftarrow k + 1$
  $\{\boldsymbol{\Theta}^k, N_k\} \leftarrow \text{LineSearch}(\lambda, \boldsymbol{\Theta}^{k-1}, L_{k-1})$
  $L_k \leftarrow \max\{L_{\min}, N_k/2\}$
until $\omega_\lambda\big(\boldsymbol{\Theta}^k\big) \leq \widehat{\epsilon}$
$\widetilde{\boldsymbol{\Theta}} \leftarrow \boldsymbol{\Theta}^k, \widehat{L} \leftarrow L_k$
return $\{\widetilde{\boldsymbol{\Theta}}, \widehat{L}\}$

Algorithm 3 $\{\boldsymbol{\Theta}, N\} \leftarrow \text{LineSearch}(\lambda, \mathbf{M}, L)$

Input $\lambda > 0, \boldsymbol{\Theta} \in \mathbb{R}^{m_1 \times m_2}, L > 0$
repeat
  $\boldsymbol{\Theta} \leftarrow \mathcal{T}_{L,\lambda}(\mathbf{M})$
  if $\phi_\lambda(\boldsymbol{\Theta}) > \psi_{L,\lambda}(\boldsymbol{\Theta}; \mathbf{M})$ **then**
    $L \leftarrow 2L$
  end if
until $\phi_\lambda(\boldsymbol{\Theta}) \leq \psi_{L,\lambda}(\boldsymbol{\Theta}; \mathbf{M})$
$N \leftarrow L$
return $\{\boldsymbol{\Theta}, N\}$

obtained by Algorithm 1, convergences at geometric rate towards $\widehat{\boldsymbol{\Theta}}$, as defined in (4.1.2).

4.2 Main Theory

In this section, we are going to present the main theoretical results for the proposed estimator in (4.1.2). We first lay out the assumptions made on the empirical loss function and the nonconvex penalty.

Suppose the SVD of $\boldsymbol{\Theta}^*$ is $\boldsymbol{\Theta}^* = \mathbf{U}^* \boldsymbol{\Gamma}^* \mathbf{V}^{*\top}$, where $\mathbf{U}^* \in \mathbb{R}^{m_1 \times r}$, $\mathbf{V}^* \in \mathbb{R}^{m_2 \times r}$ and $\boldsymbol{\Gamma}^* = \text{diag}(\boldsymbol{\gamma}_i^*) \in \mathbb{R}^{r \times r}$. We can construct the subspaces $\mathcal{F}$ and $\mathcal{F}^\perp$ as follows

$$\mathcal{F}(\mathbf{U}^*, \mathbf{V}^*) := \{\boldsymbol{\Delta} | \text{row}(\boldsymbol{\Delta}) \subseteq \mathbf{V}^* \text{ and } \text{col}(\boldsymbol{\Delta}) \subseteq \mathbf{U}^*\},$$
$$\mathcal{F}^\perp(\mathbf{U}^*, \mathbf{V}^*) := \{\boldsymbol{\Delta} | \text{row}(\boldsymbol{\Delta}) \perp \mathbf{V}^* \text{ and } \text{col}(\boldsymbol{\Delta}) \perp \mathbf{U}^*\}.$$

Shorthand notations $\mathcal{F}$ and $\mathcal{F}^\perp$ are used whenever $\mathbf{U}^*, \mathbf{V}^*$ are clear from context. It is worth noting that $\mathcal{F}$ is the span of the row and column space of $\boldsymbol{\Theta}^*$, and $\boldsymbol{\Theta}^* \in \mathcal{F}$ consequently. In addition, $\Pi_{\mathcal{F}}(\cdot)$ is the projection operator that projects matrices into the subspace $\mathcal{F}$.

To begin with, we impose two conditions on the empirical loss function $\mathcal{L}_n(\cdot)$ over a restricted set, known as restricted strong convexity (RSC) and restricted strong smoothness (RSS), respectively. Those two assumptions assume that there exist a quadratic lower bound and a quadratic upper bound, respectively, on

the remainder of the first order Taylor expansion of $\mathcal{L}_n(\cdot)$. The RSC condition has been discussed extensively in previous work [73, 65], which guarantees the strong convexity of the loss function in the restricted set and helps to control the estimation error $\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*\|_F$. In particular, we define the following subset, which is a cone of a restricted set of directions,

$$\mathcal{C} = \big\{\boldsymbol{\Delta} \in \mathbb{R}^{m_1 \times m_2} \big| \|\Pi_{\mathcal{F}^\perp}(\boldsymbol{\Delta})\|_* \leq 5\|\Pi_{\mathcal{F}}(\boldsymbol{\Delta})\|_*\big\}.$$

Assumption 4.2.1 (Restricted Strong Convexity)**.** *For operator $\mathfrak{X}$, there exists some $\kappa(\mathfrak{X}) > 0$ such that, for all $\boldsymbol{\Delta} \in \mathcal{C}$,*

$$\mathcal{L}_n(\boldsymbol{\Theta} + \boldsymbol{\Delta}) \geq \mathcal{L}_n(\boldsymbol{\Theta}) + \langle \nabla \mathcal{L}_n(\boldsymbol{\Theta}), \boldsymbol{\Delta}\rangle + \kappa(\mathfrak{X})/2\|\boldsymbol{\Delta}\|_F^2.$$

Assumption 4.2.2 (Restricted Strong Smoothness)**.** *For operator $\mathfrak{X}$, there exists some $\infty > \rho(\mathfrak{X}) \geq \kappa(\mathfrak{X})$ such that, for all $\boldsymbol{\Delta} \in \mathcal{C}$,*

$$\mathcal{L}_n(\boldsymbol{\Theta}) + \langle \nabla \mathcal{L}_n(\boldsymbol{\Theta}), \boldsymbol{\Delta}\rangle + \rho(\mathfrak{X})/2\|\boldsymbol{\Delta}\|_F^2 \geq \mathcal{L}_n(\boldsymbol{\Theta} + \boldsymbol{\Delta}).$$

Recall that $\mathcal{L}_n(\boldsymbol{\Theta}) = (2n)^{-1}\|\boldsymbol{y} - \mathfrak{X}(\boldsymbol{\Theta})\|_2$. It can be verified that with high probability $\mathcal{L}_n(\boldsymbol{\Theta})$ satisfies both RSC and RSS conditions for different applications, including matrix completion and matrix sensing. We will establish the results for RSC and RSS conditions in Section 4.2.2.

Further, we impose several regularity conditions on the nonconvex penalty $\mathcal{P}_\lambda(\cdot)$, in terms of the univariate functions $p_\lambda(\cdot)$ and $q_\lambda(\cdot)$.

Assumption 4.2.3. *(i) On the nonnegative real line, there exits a constant ν that function $p_\lambda(t)$ satisfies $p'_\lambda(t) = 0, \forall\ t \geq \nu > 0$.*

(ii) On the nonnegative real line, $q'_\lambda(t)$ is monotone and Lipschitz continuous, i.e., for $t' \geq t$, there exists a constant $\zeta_- \geq 0$ such that $q'_\lambda(t') - q'_\lambda(t) \geq -\zeta_-(t' - t)$.

(iii) Both function $q_\lambda(t)$ and its derivative $q'_\lambda(t)$ pass through the origin, i.e., $q_\lambda(0) = q'_\lambda(0) = 0$.

(iv) On the nonnegative real line, $|q'_\lambda(t)|$ is upper bounded by λ, i.e., $|q'_\lambda(t)| \leq \lambda$.

Note that condition (ii) is a type of curvature property which determines concavity level of $q_\lambda(\cdot)$, and the nonconvexity level of $p_\lambda(\cdot)$ consequently. These conditions are satisfied by many widely used nonconvex penalties, such as SCAD and MCP. For instance, it is easy to verify that SCAD penalty satisfies the conditions in Assumption 4.2.3 with $\nu = b\lambda$ and $\zeta_- = 1/(b-1)$; while for MCP, we have those conditions satisfied with

$\nu = b\lambda$ and $\zeta_- = 1/b$. Based on Assumption 4.2.2, if b is chosen such that $\kappa(\mathfrak{X}) > \zeta_-$, it can be shown that the Lipschitz constant is $L_{\widetilde{\mathcal{L}}_{n,\lambda}} = \rho(\mathfrak{X}) - \zeta_-$, and the parameter $L_{\min}$ for Algorithm 1 can be chosen such that $L_{\min} \leq \rho(\mathfrak{X}) - \zeta_-$.

4.2.1 Results for the Generic Observation Model

We first present a deterministic error bound of the estimator for the generic observation model, as stated in Theorem 4.2.4. In particular, our results implies that matrix completion via nonconvex penalty achieves a faster statistical convergence rate than the convex penalty, by taking advantage of large singular values.

Theorem 4.2.4 (Deterministic Bound for General Singular Values)**.** *Under Assumption 4.2.1, suppose that* $\widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^* \in \mathcal{C}$ *and the nonconvex penalty* $\mathcal{P}_\lambda(\boldsymbol{\Theta}) = \sum_{i=1}^m p_\lambda\big(\gamma_i(\boldsymbol{\Theta})\big)$ *satisfies Assumption 4.2.3. Under the condition that* $\kappa(\mathfrak{X}) > \zeta_-$*, for any optimal solution* $\widehat{\boldsymbol{\Theta}}$ *of (4.1.2) with regularity parameter* $\lambda \geq 2\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2/n$*, it holds that, for* $r_1 = |S_1|, r_2 = |S_2|$*,*

$$\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*\|_F \leq \underbrace{\frac{\tau\sqrt{r_1}}{\kappa(\mathfrak{X}) - \zeta_-}}_{S_1:\boldsymbol{\gamma}_i^* \geq \nu} + \underbrace{\frac{3\lambda\sqrt{r_2}}{\kappa(\mathfrak{X}) - \zeta_-}}_{S_2:\nu > \boldsymbol{\gamma}_i^* > 0}, \tag{4.2.1}$$

where $\tau = \big\|\boldsymbol{\Pi}_{\mathcal{F}_{S_1}}\big(\nabla\mathcal{L}_n(\boldsymbol{\Theta}^*)\big)\big\|_2$*, where* $\mathcal{F}_{S_1}$ *is a subspace of* $\mathcal{F}$ *associated with* S_1*.*

It is important to note that the upper bound on the Frobenius norm-based estimation error includes two parts corresponding to different magnitudes of the singular values of the true matrix, *i.e.*, $\boldsymbol{\gamma}_i^*$: (i) S_1 corresponds to the set of singular values with larger magnitudes; and (ii) S_2 corresponds to the set of singular values with smaller magnitudes. By setting $\zeta_- = \kappa(\mathfrak{X})/2$, we have

$$\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*\|_F \leq 2\tau\sqrt{r_1}/\kappa(\mathfrak{X}) + 6\lambda\sqrt{r_2}/\kappa(\mathfrak{X}).$$

We can see that provided that $r_1 > 0$, the rate of the proposed estimator is faster than the nuclear norm based one, i.e, $\mathcal{O}\big(\lambda\sqrt{r}/\kappa(\mathfrak{X})\big)$[71], in light of the fact that $\tau = \big\|\boldsymbol{\Pi}_{\mathcal{F}_{S_1}}\big(\nabla\mathcal{L}_n(\boldsymbol{\Theta}^*)\big)\big\|_2$ is order of magnitude smaller than $\big\|\nabla\mathcal{L}_n(\boldsymbol{\Theta}^*)\big\|_2 = \lambda$. This would be demonstrated in more detail for specific examples, *i.e.*, matrix completion and matrix sensing, in Section 4.2.2. In particular, if $\boldsymbol{\gamma}_r^* \geq \nu$, meaning that all the nonzero singular values are larger than ν, the proposed estimator attains the best-case convergence rate of $2\tau\sqrt{r}/\kappa(\mathfrak{X})$.

In Theorem 4.2.4, we have shown that the convergence rate of nonconvex penalty based estimator is faster than the nuclear norm based one. In the following, we show that under certain assumptions on the

magnitudes of the singular values, the estimator in (4.1.2) enjoys the oracle properties, namely, the obtained M-estimator performs as well as if the underlying model were known beforehand. Before presenting the results on the oracle property, we first formally introduce the oracle estimator,

$$\widehat{\mathbf{\Theta}}_O = \underset{\mathbf{\Theta} \in \mathcal{F}(\mathbf{U}^*, \mathbf{V}^*)}{\operatorname{argmin}} \mathcal{L}_n(\mathbf{\Theta}). \tag{4.2.2}$$

Remark that the objective function in (4.2.2) only includes the empirical loss term because the optimization program is constrained in the rank-r subspace $\mathcal{F}(\mathbf{U}^*, \mathbf{V}^*)$. Since it is impossible to get $\mathbf{U}^*, \mathbf{V}^*$ and the rank r in practice, *i.e.*, $\mathcal{F}(\mathbf{U}^*, \mathbf{V}^*)$ is unknown, the oracle estimator defined above is not a practical estimator. We analyze the estimator in (4.1.2) when $\kappa(\mathfrak{X}) > \zeta_-$, under which condition $\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) = \mathcal{L}_n(\mathbf{\Theta}) + \mathcal{P}_\lambda(\mathbf{\Theta})$ is strongly convex over the restricted set $\mathcal{C}$ and $\widehat{\mathbf{\Theta}}$ is the unique global optimal solution for the optimization problem. Moreover, the following theorem shows that under suitable conditions, the estimator in (4.1.2) is identical to the oracle estimator.

Theorem 4.2.5 (Oracle Property)**.** *Under Assumption 4.2.1 and 4.2.2, suppose that* $\widehat{\mathbf{\Delta}} = \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^* \in \mathcal{C}$ *and* $\mathcal{P}_\lambda(\mathbf{\Theta}) = \sum_{i=1}^r p_\lambda(\gamma_i(\mathbf{\Theta}))$ *satisfies regularity condition (i), (ii), (iii) in Assumption 4.2.3. If* $\kappa(\mathfrak{X}) > \zeta_-$ *and* $\boldsymbol{\gamma}^*$ *satisfies the condition that*

$$\min_{i \in S} \left|(\boldsymbol{\gamma}^*)_i\right| \geq \nu + \frac{2\sqrt{r}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2}{n\kappa(\mathfrak{X})}, \tag{4.2.3}$$

where $S = \operatorname{supp}(\boldsymbol{\gamma}^*)$*. For the estimator in (4.1.2) with choice of regularization parameter* $\lambda \geq 2n^{-1}\left\|\mathfrak{X}^*(\boldsymbol{\epsilon})\right\|_2 + 2n^{-1}\sqrt{r}\rho(\mathfrak{X})\left\|\mathfrak{X}^*(\boldsymbol{\epsilon})\right\|_2/\kappa(\mathfrak{X})$*, we have that* $\widehat{\mathbf{\Theta}} = \widehat{\mathbf{\Theta}}_O$*, indicating* $\mathit{rank}(\widehat{\mathbf{\Theta}}) = \mathit{rank}(\widehat{\mathbf{\Theta}}_O) = \mathit{rank}(\mathbf{\Theta}^*) = r$*. Moreover, we have,*

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq 2\sqrt{r}\tau/\kappa(\mathfrak{X}), \tag{4.2.4}$$

where $\tau = \left\|\Pi_{\mathcal{F}}\big(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\big)\right\|_2$.

Theorem 4.2.5 implies that, with a suitable choice of regularization parameter λ, if the magnitude of the smallest nonzero singular value is sufficiently large, *i.e.*, satisfying (4.2.3), the proposed estimator in (4.1.2) is identical to the oracle estimator. This is a very strong result because we do not even know the subspace $\mathcal{F}$. The direct consequence is that the M-estimator exactly recovers the rank of the true matrix, $\mathbf{\Theta}^*$. Moreover, as Theorem 4.2.5 is a specific case of Theorem 4.2.4 with $r_1 = r$, we immediately have that the convergence rate in Theorem 4.2.5 corresponds to the best-case convergence rate in (4.2.1), which is identical to the statistical rate of the oracle estimator.

4.2.2 Results for Specific Examples

The deterministic results in Theorem 4.2.4 and Theorem 4.2.5 are fairly abstract in nature. In what follows, we consider the two specific examples of low-rank matrix estimation as in Section 4.1.2, and show how the results obtained so far yield concrete and interpretable results. More importantly, we rigorously demonstrate the improvement of the proposed estimator on statistical convergence rate over the traditional one with nuclear norm penalty. More results on oracle property can be found in Appendix, Section B.5.

Matrix Completion

We first analyze the example of matrix completion, as discussed earlier in Example 4.1.1. It is worth noting that under a suitable condition on spikiness ratio[1], we can establish the restricted strongly convexity, as stated in Assumption 4.2.1.

Corollary 4.2.6. *Suppose that $\widehat{\mathbf{\Delta}} = \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^* \in \mathcal{C}$, the nonconvex penalty $\mathcal{P}_\lambda(\mathbf{\Theta})$ satisfies Assumption 4.2.3, and $\mathbf{\Theta}^*$ satisfies spikiness assumption, i.e., $\|\mathbf{\Theta}^*\|_\infty \le \alpha^*$, then for any optimal solution $\widehat{\mathbf{\Theta}}$ to the slight modification of (4.1.2), i.e.,*

$$\widehat{\mathbf{\Theta}} = \underset{\mathbf{\Theta}\in\mathbb{R}^{m_1\times m_2}}{\operatorname{argmin}} \frac{1}{2n}\big\|\boldsymbol{y} - \mathfrak{X}(\mathbf{\Theta})\big\|_2^2 + \mathcal{P}_\lambda(\mathbf{\Theta}),$$
$$\text{subject to} \quad \|\mathbf{\Theta}\|_\infty \le \alpha^*,$$

there are universal constants $C_1, \ldots, C_5$, with regularity parameter $\lambda \ge C_3\sigma\sqrt{\log M/(nm)}$ and $\kappa = C_4/(m_1m_2) > \zeta_-$, it holds with probability at least $1 - C_5/M$ that

$$\frac{1}{\sqrt{m_1m_2}}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \le \max\{\alpha^*, \sigma\}\left[C_1r_1\sqrt{\frac{\log M}{n}} + C_2\sqrt{\frac{r_2M\log M}{n}}\right].$$

Remark 4.2.7. *Corollary 4.2.6 is a direct result of Theorem 4.2.4. Recall the convergence rate[2] of matrix completion with nuclear norm penalty due to [53, 34], which is as follows*

$$\frac{\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F}{\sqrt{m_1m_2}} = \mathcal{O}\left(\max\{\alpha^*, \sigma\}\sqrt{\frac{rM\log M}{n}}\right). \tag{4.2.5}$$

It is evident that if $r_1 > 0$, i.e., we have r_1 singular values that are larger than ν, the convergence rate obtained by a nonconvex penalty is faster than the one obtained with the convex penalty. In the worst case,

[1] It is insufficient to recover the low-rank matrices due to its infeasibility of recovering overly "spiky" matrices which has very few large entries. Additional assumption on spikiness ratio is needed. Details on spikiness are given in Appendix, Section B.5.1.

[2] Similar statistical convergence rate was obtained in [72] for nonuniform sampling schema.

when all the singular values are smaller than ν, our result reduced to (4.2.5) with $r_2 = r$. Meanwhile, if the magnitude of singular values satisfies the condition that $\min_{i\in S}\gamma_i^* \geq v$, i.e., $r_1 = r$ ($S_1 = S$), the convergence rate of our results is $\mathcal{O}\big(\sqrt{r^2\log M/n}\big)$. In [53, 72], the authors proved a minimax lower bound for matrix completion, which is $O(\sqrt{rM/n})$. Our result is not contradictory to the minimax lower bound, because the lower bound is proved for the general class of low rank matrices, while our result takes advantage of the large singular values. In other words, we consider a specific (potentially smaller) class of low rank matrices with both large and small singular values.

Matrix Sensing With Dependent Sampling

In the example of matrix sensing, a more general model with dependence among the entries of $\mathbf{X}_i$ is considered. Denote $\mathrm{vec}(\mathbf{X}_i) \in \mathbb{R}^{m_1m_2}$ as the vectorization of $\mathbf{X}_i$. For a symmetric positive definite matrix $\mathbf{\Sigma} \in \mathbb{R}^{m_1m_2\times m_1m_2}$, it is called $\mathbf{\Sigma}$-Ensemble [71] if the elements of observation matrices $\mathbf{X}_i$'s are sampled from $\mathrm{vec}(\mathbf{X}_i) \sim N(\mathbf{0}, \mathbf{\Sigma})$. Define $\pi^2(\mathbf{\Sigma}) = \sup_{\|\mathbf{u}\|_2=1,\|\mathbf{v}\|_2=1}\mathrm{Var}(\mathbf{u}^\top\mathbf{X}\mathbf{v})$, where $\mathbf{X} \in \mathbb{R}^{m_1\times m_2}$ is a random matrix sampled from the $\mathbf{\Sigma}$-Ensemble. Specifically, when $\mathbf{\Sigma} = \mathbf{I}$, it can be verified that $\pi(\mathbf{I}) = 1$, corresponding to the classical matrix sensing model where the entries of $\mathbf{X}_i$ are independent from each other.

Corollary 4.2.8. *Suppose that $\widehat{\mathbf{\Delta}} = \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^* \in \mathcal{C}$ and the nonconvex penalty $\mathcal{P}_\lambda(\mathbf{\Theta})$ satisfies Assumption 4.2.3, if the random design matrix $\mathbf{X}_i \in \mathbb{R}^{m_1\times m_2}$ is sampled from the $\mathbf{\Sigma}$-ensemble and $\lambda_{\min}(\mathbf{\Sigma})$ is the minimal eigenvalue of $\mathbf{\Sigma}$, there are universal constants $C_1, \ldots, C_6$, such that, if $\kappa(\mathfrak{X}) = C_3\lambda_{\min}(\mathbf{\Sigma}) > \zeta_-$ for any optimal solution $\widehat{\mathbf{\Theta}}$ of (4.1.2) with $\lambda \geq C_4\sigma\pi(\mathbf{\Sigma})\big(\sqrt{m_1/n} + \sqrt{m_2/n}\big)$, it holds with probability at least $1 - C_5\exp\big(-C_6(m_1+m_2)\big)$ that*

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq \frac{\sigma\pi(\mathbf{\Sigma})}{\lambda_{\min}(\mathbf{\Sigma})\sqrt{n}}\big[C_1r_1 + C_2\sqrt{r_2M}\big].$$

Remark 4.2.9. *Similarly, Corollary 4.2.8 is a direct consequence of Theorem 4.2.4. The problem has been studied by [71] via convex relaxation, with the following estimator error bound*

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F = \mathcal{O}\bigg(\frac{\sigma\pi(\mathbf{\Sigma})\sqrt{rM}}{\lambda_{\min}(\mathbf{\Sigma})\sqrt{n}}\bigg). \tag{4.2.6}$$

When there are $r_1 > 0$ singular values that are larger than ν, the result obtained in Corollary 4.2.8 implies that the convergence rate of the proposed estimator is faster than (4.2.6). When $r_1 = r$, we obtain the best-case convergence rate of $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F = \mathcal{O}\big(\sigma\pi(\mathbf{\Sigma})r/(\sqrt{n}\lambda_{\min}(\mathbf{\Sigma}))\big)$. In the worst case, when $r_1 = 0$ and $r_2 = r$, the results in Corollary 4.2.8 reduce to (4.2.6).

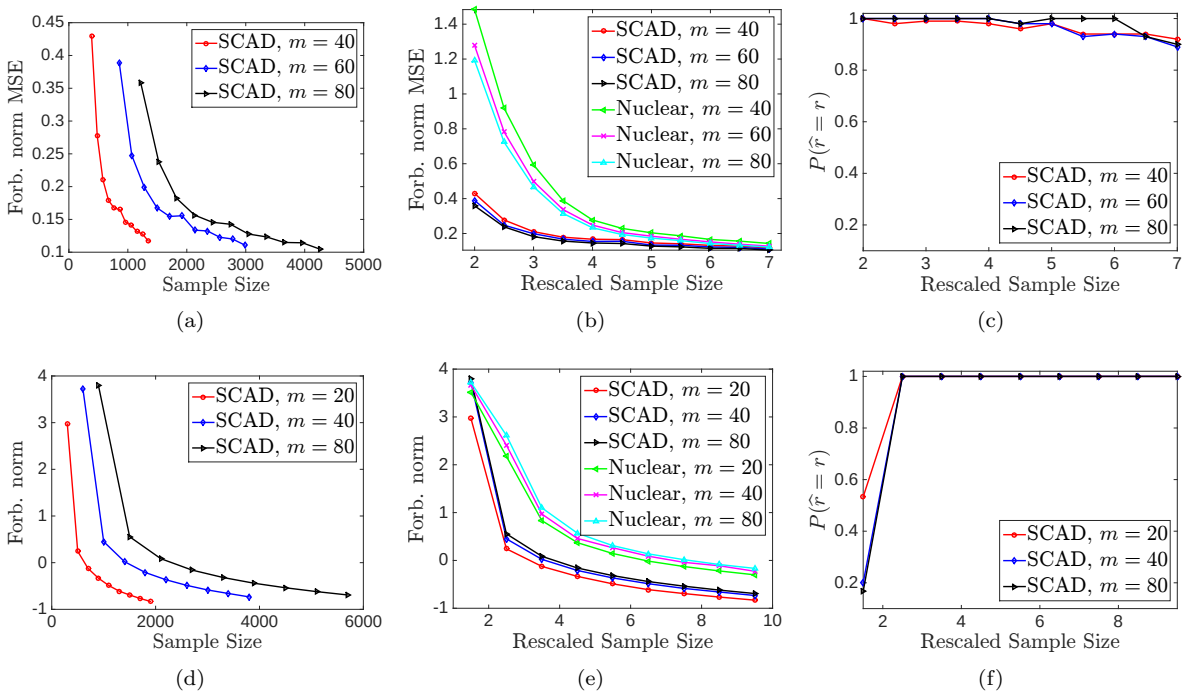


Figure 4.1: Simulation Results for Matrix Completion and Matrix Sensing with SCAD penalty. The size of matrix is $m \times m$.

4.3 Numerical Experiments

In this section, we study the performance of the proposed estimator by various simulations and numerical experiments on real-word datasets. It it worth noting that we study the proposed estimator with $\zeta_- < \kappa(\mathfrak{X})$, which can be attained by setting $b = 1 + 2/\kappa(\mathfrak{X})$ for the SCAD penalty. Similarly, the parameter for MCP penalty can be set that $b = 2/\kappa(\mathfrak{X})$.

4.3.1 Simulations

The simulation results demonstrate the close agreement between theoretical upper bound and the numerical behavior of the M-estimator. Simulations are performed for both matrix completion and matrix sensing. In both cases, we solved instances of optimization problem (4.1.2) for a square matrix $\mathbf{\Theta}^* \in \mathbb{R}^{m \times m}$. For $\mathbf{\Theta}^*$ with rank r, we generate $\mathbf{\Theta}^* = \mathbf{ABC}^\top$, where $\mathbf{A}, \mathbf{C} \in \mathbb{R}^{m \times m}$ are the left and right singular vectors of a random matrix, and set $\mathbf{B}$ to be a diagonal matrix with r nonzero entries, and the magnitude of each nonzero entries is above $\nu = \lambda b$, *i.e.*, $r_1 = r$. The regularization parameter λ is chosen based on theoretical results with σ^2 assumed to be known.

In the following, we report detailed results on the estimation errors of the obtained estimators and the

Table 4.1: Results on image recovery in terms of RMSE ($\times 10^{-2}$, mean $\pm$ std).

Image	SVP	SoftImpute	AltMin	TNC	R1MP	Nuclear	SCAD	MCP
Lenna	3.84 ± 0.02	4.58 ± 0.02	4.43 ± 0.11	5.49 ± 0.62	3.91 ± 0.03	5.05 ± 0.17	2.79 ± 0.02	2.81 ± 0.04
Barbara	4.49 ± 0.04	5.23 ± 0.03	5.05 ± 0.05	6.57 ± 0.92	4.71 ± 0.06	6.48 ± 0.53	4.74 ± 0.02	4.73 ± 0.03
Clown	3.75 ± 0.03	4.43 ± 0.05	5.44 ± 0.41	6.92 ± 1.89	3.89 ± 0.05	3.70 ± 0.24	2.77 ± 0.01	2.81 ± 0.01
Crowd	4.49 ± 0.04	5.35 ± 0.07	4.78 ± 0.09	7.44 ± 1.23	4.88 ± 0.06	4.44 ± 0.18	3.64 ± 0.07	3.68 ± 0.09
Girl	3.35 ± 0.03	4.12 ± 0.03	5.01 ± 0.66	4.51 ± 0.52	3.06 ± 0.02	4.77 ± 0.34	2.06 ± 0.01	2.05 ± 0.02
Man	4.42 ± 0.04	5.17 ± 0.03	5.17 ± 0.17	6.01 ± 0.62	4.61 ± 0.03	5.44 ± 0.45	3.42 ± 0.04	3.40 ± 0.02

Table 4.2: Recommendation results measured in term of the averaged RMSE.

dataset	SVP	SoftImpute	AltMin	TNC	R1MP	Nuclear	SCAD	MCP
Jester1	4.7318	5.1211	4.8562	4.4803	4.3401	4.6910	4.1721	4.1719
Jester2	4.7712	5.1523	4.8712	4.4511	4.3721	4.5597	4.2002	4.1987
Jester3	8.7439	5.4532	9.5230	4.6712	4.9803	5.1231	4.6729	4.6740

probability of exactly recovering the true rank (oracle property). Due to space limitation, we include the simulation results using MCP in the appendix.

Matrix Completion. We study the performance of estimators with both convex and nonconvex penalties for $m \in \{40, 60, 80\}$, and the rank $r = \lfloor \log^2 m \rfloor$. $\mathbf{X}_i$'s are uniformed sampled over $\mathcal{X}$, with the variance of observation noise $\sigma^2 = 0.25$. For every configuration, we repeat 100 trials and compute the averaged mean squared Frobenius norm error $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F^2/m^2$ over all trials.

Figure 4.1(a)-4.1(c) summarize the results for matrix completion. Particularly, Figure 4.1(a) plots the mean-squared Frobenius norm error versus the raw sample size, which shows the consistency that estimation error decreases when sample size increases, while Figure 4.1(b) plots the MSE against the *rescaled sample size* $N = n/(rm \log m)$. It is clearly shown in Figure 4.1(b) that, in terms of estimation error, the proposed estimator with SCAD penalty outperforms the one with nuclear norm, which aligns with our theoretical analysis. Finally, the probability of exactly recovering the rank of underlying matrix is plotted in Figure 4.1(c), which indicates that with high probability the rank of underlying matrix can be exactly recovered.

Matrix Sensing. For matrix sensing, we set the rank $r = 10$ for all $m \in \{20, 40, 80\}$. $\mathbf{\Theta}^*$ is generated similarly as in matrix completion. We set the observation noise variance $\sigma^2 = 1$ and $\mathbf{\Sigma} = \mathbf{I}$, *i.e.*, the entries of $\mathbf{X}_i$ are independent. Each setting is repeated for 100 times.

Figure 4.1(d)-4.1(f) correspond to results of matrix sensing. The Frobenius norm $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F$ is reported in log scale. Figure 4.1(d) demonstrate how the estimation errors scale with m and n, which aligns well with our theoretical findings. Also, as observed in Figure 4.1(e), the estimator with SCAD penalty has lower error bounds compared with the one of nuclear norm penalty. At last, it shows in Figure 4.1(f) that, empirically, the underlying rank is perfectly recovered by the nonconvex estimator when n is sufficiently large ($n \geq 3rm$).

4.3.2 Experiments on Real World Datasets

In this section, we apply our proposed matrix completion estimator to two real-world applications, image inpainting and collaborative filtering, and compare it with some existing methods, including singular value projection (SVP) [41], nuclear norm Constraint (TNC) [40], alternating minimization (AltMin) [43], spectral regularization algorithm (SoftImpute) [67], rank-one matrix pursuit (R1MP) [109], and nuclear norm penalty [71].

Image Inpainting We select 6 images [3] to test the performance of different algorithms. The matrices corresponding to selected images are of the size 512×512. We project the underlying matrices into the corresponding subspaces associated with the top $r = 200$ singular values of each matrix, by which we can guarantee that the problem being solved is a low-rank one. In addition, we randomly select 50% of the entries as observations. Each trial is repeated 10 times. The performance is measured by *root mean square error* (RMSE) [40, 83], summarized in Table 4.1. As shown in Table 4.1, the estimators obtained with nonconvex penalties, including SCAD penalty and MCP, achieve the best performance, and significantly outperform the other algorithms on all pictures, except for Barbara. It is worth noting that due to the similar properties of MCP and SCAD, the results of SCAD and MCP are comparable. Moreover, the estimators with nonconvex penalties have smaller RMSE for all pictures, compared with the nuclear norm based estimator, which backs up our theoretical analysis, and the improvement is significant compared with some specific algorithms.

Collaborative Filtering Considering the matrix completion algorithms for recommendations, we demonstrate using three datasets: Jester1[4], Jester2 and Jester3, which contain rating data of users on jokes, with real-valued rating scores ranging from -10.0 to 10.0. The sizes of these matrices are $\{24983, 23500, 24983\} \times 100$, containing 10^6, 10^6, 6×10^5 ratings, respectively. We randomly select 50% of the ratings as observations, and make predictions over the remaining 50%. Each run is repeated for 10 times. According to the numerical results summarized in Table 4.2, we observe that the proposed estimators (SCAD, MCP) have the best performance among all existing algorithms. In particular, the estimator with nonconvex penalties (*i.e.*, MCP, SCAD) is better than the estimator with nuclear norm penalty, which agrees well with the results obtained. Comparable results of MCP and SCAD are observed.

[3] The images can be downloaded from http://www.utdallas.edu/~cxc123730/mh_bcs_spl.html.
[4] The Jester dataset can be downloaded from http://eigentaste.berkeley.edu/dataset/.

Chapter 5

Embedding Learning in Heterogeneous Information Networks with Events

5.1 Preliminaries

In this section, we define the problem of embedding learning with events in heterogeneous information networks and introduce several related concepts and necessary notations.

5.1.1 Heterogeneous Information Networks and Events

Definition 5.1.1 (Information Networks)**.** *Given a set of objects belonging to T types $\mathcal{X} = \{X_t\}_{t=1}^{T}$, where X_t represents the set of distinct objects with t-th type, a network $G = \langle \mathcal{X}, \mathcal{E} \rangle$ is called* ***an information network on objects*** *$\mathcal{X}$, where $\mathcal{E}$ is a set of binary interactions of objects in $\mathcal{X}$. Specifically, such an information network is called a* ***heterogeneous information network*** *if $T \geq 2$; and* ***homogeneous information network*** *if $T = 1$.*

Regarding the binary interactions of $\mathcal{E}$ on objects $\mathcal{X}$, we define an event as a set of objects having interactions that happen simultaneously.

Definition 5.1.2. *(Events) For a set of objects $V_i \subset \mathcal{X}$, if a set of interactions among objects in V_i happen at the same time, we define it as an event Q_i represented by a triplet $\langle q_i, V_i, \omega_i \rangle$, where q_i is the event identifier serving as the index of the event, $V_i \subseteq \mathcal{X}$ is the set of participating objects and ω_i is the weight of event Q_i (e.g., the number of occurrences of this event).*

In later sections we will slightly abuse notation and use X_t to indicate both the set of objects belonging to t-th type and the name of the type as well. Besides multiple object types, we also study the general case with multiple event types where each event type is associated with an *event schema*, which is a tool to visualize relationships among the event and participating objects. The network schema of DBLP of Example 1.3.1 is shown on the left of Figure 1.2 with one event type. Yelp data described in the following example contain two event types. Event identifiers are marked in dashed circles.

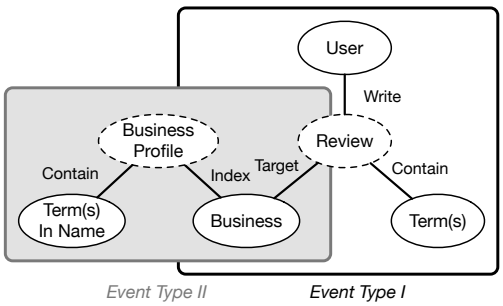


Figure 5.1: Event schema of Yelp with two event types, business profile (left) and review (right).

Example 5.1.3. *Yelp (http://www.yelp.com/) is an online website for users to review various businesses, which can be naturally represented as a heterogeneous information network. Based on schema shown in Figure 5.1, there are two types of events. The first event type (left) is business profile, the participating object types of which include Terms in Name and Business; The second (right) is the review event, including User, Business, and Term types. The business objects type participates in both event types.*

5.1.2 Learning Object Embedding

Given a heterogeneous information network and the event schema, embedding algorithms learn to represent each object of different types using a low-dimensional vector in the same space. The embedding algorithms are to preserve the semantic similarity among objects as well as event topological structures such that objects that are semantically similar and co-occur in the same event will be close in the space, with the distance measured by cosine similarity, for instance.

Object embedding learning for heterogeneous information networks has broad applications [92, 21]. A straightforward method would ignore the object types and learn the object embeddings using network models, such as LINE [93]. In existing work, heterogeneous event data are modeled as a heterogeneous information network, in which the objects are of mutliple types and the relations between objects are also of multiple types [89]. [92, 21] decompose each event into multiple *binary* relations between objects. Object embeddings are thus learned based on all sets of binary relations independently in a joint optimization framework. However, as we discussed in Section **??**, such a decomposition method may lose some subtle information within the heterogeneous information networks with event property and lead to information loss.

Instead of simply considering each event as a set of independent binary relations between individual participating objects, we define a new structure to encapsulate all the information based on the events. Inspired from classical analysis on hypergraphs and hyperedges [6, 84], for each event Q_i, we use a corresponding

hyperedge H_i to model the event by viewing all the participating objects as a whole in the hyperedge, *i.e.*, H_i with q_i as its identifier connects the set of objects V_i with edge (event) weight ω_i. When the number of participating objects in each event is two, our model reduces to classical network model with binary interactions. Therefore, each event corresponds to one binary interaction.

In order to model the semantic similarity among participating objects in each event, which consequently preserves the topological structures of events, we propose two methods based on different semantics of prediction. The first insight is that semantically related objects are more likely to participate in the same event. For instance, in the DBLP data, it is more frequently to observe publications with author "Christos Faloutsos" and term "Network" in the venue ICDM. Therefore, we define the object-driven proximity based on the prediction of participating object observation given the other participating objects as context.

Definition 5.1.4. *The **Object-driven proximity** of an event is defined as the likelihood of observing a target object given all other participating objects on the same hyperedge corresponding to an event.*

Based on Object-Driven Proximity, we aim at predicting a target object. Therefore, the corresponding embedding algorithm is called HEBE **P**redict **O**bject (HEBE-PO) where HEBE stands for hyperedge-based embedding.

Since the HEBE-PO approach considers only the semantic proximity among participating objects on the same event (i.e., hyperedge), we further take the event itself into consideration. The second approach therefore is to predict the event given the set of participating objects. In other words, we additionally assign embeddings to each hyperedge (through event identifier) so that the proximity is well-defined, as follows.

Definition 5.1.5. *The **hyperedge-driven proximity** is defined as the likelihood of observing a hyperedge given all the participating objects.*

For example, given the set of author (Christos Faloutsos), term (Network) and venue (ICDM), we additionally learn the embedding of the publication event record, and connect the set of participating objects with the right publication record as an observed hyperedge. The corresponding embedding algorithm is called HEBE **P**redict Hyper**E**dge (HEBE-PE). The underlying intuition of HEBE-PO is that the semantic meanings of all participating objects are close to the semantic meaning of the event. Particularly, the event embedding serves as summarization of all the participating objects, which can effectively filter out noise information within each event if exists.

Based on the two kinds of proximity that preserves the event structures as defined above, we formally define the task of object embedding as follows.

Definition 5.1.6 (Object Embedding for Heterogeneous Information Network)**.** *Given a heterogeneous information network, represented as a collection of events $\mathcal{D} = \{Q_i\}$, and the event schema,* ***object embedding*** *is to learn a function $\mathcal{M}$ that projects each object to a vector in a d-dimension space $\mathbb{R}^d$ that keeps either object-driven or hyperedge-driven proximity, where $d \ll |\mathcal{X}|$, i.e., $\mathcal{M} : \mathcal{X} \to \mathbb{R}^d$, where $\mathcal{X}$ is the set of all objects.*

5.2 Hebe Framework

In this section, we introduce the Hebe framework to learn the object embeddings. The major difficulty that lies in embedding learning in heterogeneous event data is how to model and optimize the proximity among participating objects in each event. We will provide the details of estimating the proximity as introduced in Section 5.1, and the optimization of which will be discussed in Section 5.3.

5.2.1 SubEvent Sampling

Before diving into the object learning, we first introduce a concept of *SubEvent Sampling* to simplify the representation of heterogeneous events. Recall that for an event Q_i, we represent it as $\langle q_i, V_i, \omega_i \rangle$, where $V_i \subset \mathcal{X}$. Particularly, we denote $V_i^t \subset V_i$ as the set of objects in type X_t.

In real world scenario, we have that $|V_i^t| \geq 1$. For the case when $|V_i^t| > 1$, we need to additionally include the multiple objects in each type. For instance, in DBLP (as discussed in Example 1.3.1), for each publication event, we have only one venue but multiple authors and terms. Also, in Yelp (Example 5.1.3), each review event is about one business but the review text may contain more than one hundred terms. Therefore, there is a count imbalance problem for $|V_i^t|$ with $t = 1, \ldots, T$.

To address the imbalance problem, we propose to sample subevent from each event by uniformly sampling one object from each object type. For instance, given an event of $Q_i = \langle q_i, V_i, \omega_i \rangle$, we have $V_i = \{a_1, a_2\} \cup \{t_1, t_2, t_3\} \cup \{v_1\}$ (where $a.$, $t.$ and $v.$ stand for author, term, and venue objects, respective, as shown in Figure 5.2), we can sample a subevent $Q_{i,s} = \{a_2, t_2, v_1\}$ with probability of $1/(2 \times 3)$, consequently, we assign the weight for $Q_{i,s}$ as $\omega_i/(2 \times 3)$.

For a more general case of Q_i, we can sample $S_i = \prod_{t=1}^{T} |V_i^t|$ subevents, with the weight of each subevent as ω_i/S_i. Notably, we can see for each object $v_{i,t} \in V_i^t$, the aggregated weight for the object is $\omega_i/|V_i^t|$, which naturally balances the number of objects in each type. In other words, the more objects are in one type, the less important each object of that type is, vice versa. In addition, for each Q_i, it can be *losslessly* recovered by $\{Q_{i,s}\}_{s=1}^{S_i}$.

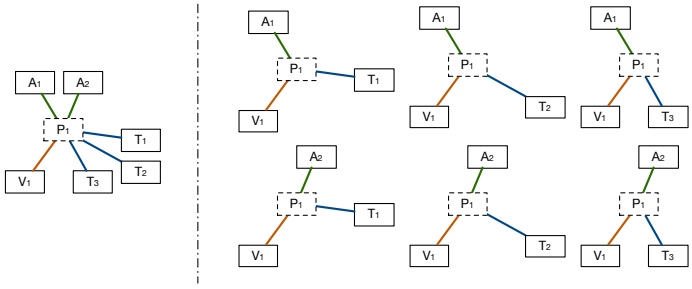


Figure 5.2: Illustration of SubEvent Sampling. The event (on the left) has 6 subevents (on the right). The weight of each subevent is the same regarding the event.

We denote $\widetilde{Q}_{i'} = Q_{i,s}$ for $i = 1, \ldots, n$, $s = 1, \ldots, S_i$ with i' obtained sequentially, as one subevent w.r.t. the event Q_i. We flatten the subevent structures within events by defining $\widetilde{\mathcal{Q}} = \left\{\widetilde{Q}_{i'} = \langle \widetilde{q}_{i'}, \widetilde{V}_{i'}, \widetilde{\omega}_{i'} \rangle\right\}_{i'=1}^{N}$ with $N = \sum_{i=1}^{n} \prod_{t=1}^{T} |V_i^t|$, where the weight of $\widetilde{Q}_{i'}$ is $\widetilde{\omega}_{i'} = \omega_i / S_i$, since $\widetilde{Q}_{i'}$ is a subevent sampled from Q_i. It is worth noting that different events could generate the same subevent. For instance, both $Q_1 = \{a_1, a_2\} \cup \{t_1, t_2, t_3\} \cup \{v_1\}$ and $Q_2 = \{a_1, a_3\} \cup \{t_1, t_4, t_5\} \cup \{v_1\}$ have $\widetilde{Q}_1 = \{a_1, t_1, v_1\}$ as a subevent.

For implementation, we do not need to generate $\widetilde{\mathcal{Q}}$ (the set of all subevents), since $|\widetilde{\mathcal{Q}}| \gg |\mathcal{Q}|$. Our stragety is to do subevent sampling on the fly. Suppose event Q_i is sampled with probability proportional to ω_i, we randomly sample one object from each object type and obtain a subevent $\widetilde{Q}_{i'}$. Specifically, the probability of a subevent $\widetilde{Q}_{i'}$ being sampled is propotional to ω_i / S_i as desired. The probability of subevents from more than one event can be aggregated accordingly.

Thereafter, for the following analysis we use event and subevent interchangably and we drop $\widetilde{\ }$ for $\widetilde{Q}, \widetilde{q}, \widetilde{V}, \widetilde{\omega}$ whenever the context is clear. We can also use hyperedges to encapsulate subevent topological structures. Without loss of generality, we assume there are no duplicated subevents $\widetilde{Q}_{i'}$'s and the weight $\widetilde{\omega}_{i'}$for each subevent $\widetilde{Q}_{i'}$ has been aggregated appropriately.

5.2.2 Hebe Object Prediction

As defined in Definition 5.1.4, HEBE-PO is to predict a target object out of all alternative objects given the other participating objects on the same hyperedge as context. Due to the heterogeneity of the objects, we constrain that the alternative objects are of the same type as the target object. When the target object type X_t is given, the corresponding target object in each subevent is accordingly $u_t \in V_i^t$ where $|V_i^t| = 1$ for each subevent. Without loss of generality, we further assume the target object is of type X_1. We denote the target object as u, context object set as C. Obviously, $|C| = T - 1$ for subevents and $u \notin C$. The conditional

probability of predicting the target object u is defined as

$$\mathbb{P}_o(u|C) = \frac{\exp\big(S(u,C)\big)}{\sum_{v\in X_1}\exp\big(S(v,C)\big)}, \tag{5.2.1}$$

where $S(\cdot)$ is a scoring function reflecting the similarity between target object u and context objects C. Intuitively, (5.2.1) can be understood as given C selecting u from the pool of candidates X_1. Suppose $C = \{c_2, c_3, \ldots, c_T\}$, we have the scoring function defined as follows:

$$S(u,C) = \big\langle \mathbf{w}_u, \frac{1}{T-1}\sum_{t=2}^{T}\mathbf{w}_{c_t}\big\rangle, \tag{5.2.2}$$

where $\mathbf{w}_u \in \mathbb{R}^d$ is the embeddings of u.

We remark that the choice of scoring function is a free parameter and can be altered based on specific applications. The HEBE framework does not depend on the choice of the scoring function.

It is worth noting that we learn one embedding for objects of each type t. However, if there are nodes of the same type playing different roles in the event schema, we learn embeddings for the objects w.r.t. each role. For instance, in the word occurrence event, we learn two embeddings for the set of word objects, corresponding to context and target, as in [68, 93]. On the other hand, if one type of objects appears in two event schemata, we only learn one embedding for each object in that type and the semantics of the shared objects are the same in the events of different types.

Objective. Suppose the target object type t. To preserve the object-driven proximity, we can naturally minimize Kullback-Leibler (KL) divergence between model distribution $\mathbb{P}_o(\cdot|C_t)$ and empirical distribution $\widehat{\mathbb{P}}_o(\cdot|C_t)$ where C_t is an arbitrary choice of context. We additionally define $\mathcal{P}_t$ as the sample space of $C_{i,t}$ for $i = 1, \ldots, N$, such that $\mathcal{P}_t = \{C_{i,t}\}_{i=1}^N$ is the collection of possible values of context C in the empirical observations of $\mathcal{Q}$. Therefore, the objective function is:

$$\mathcal{L}_o = -\sum_{t=1}^{T}\sum_{C_t\in\mathcal{P}_t}\lambda_{C_t}\mathrm{KL}\Big(\widehat{\mathbb{P}}_o(\cdot|C_t), \mathbb{P}_o(\cdot|C_t)\Big),$$

where we use λ_{C_t} is the importance of the context C_t, $\lambda_{C_t} = \sum_{i=1}^N \omega_i \mathbf{I}_{\{C_t = V_i\setminus u_{i,t}\}}$, where $u_{i,t} \in V_i$ is the target object in type X_t and $\mathbf{I}_{\{\cdot\}}$ is a binary indicator function. λ_{C_t} can be intuitively understood as the weighted number of subevent hyperedges that have C_t as context.

Note that we assign the same weight to each object type. The model can be further extended to

distinguish the relative importance for different types, as follows:

$$\mathcal{L}'_o = -\sum_{t=1}^{T} \gamma_t \cdot \sum_{C_t \in \mathcal{P}_t} \lambda_{C_t} \mathrm{KL}\Big(\widehat{\mathbb{P}}_o(\cdot|C_t), \mathbb{P}_o(\cdot|C_t)\Big),$$

with γ_t as importance parameter of object type X_t. For simplicity, we set $\gamma_1 = \cdots = \gamma_T = 1$. We leave the more general cases with different γ_t's for future work.

Lemma 5.2.1. *Maximizing $\mathcal{L}_o$ is equivalent to maximizing*

$$L_o = \sum_{t=1}^{T} \sum_{i=1}^{N} \omega_i \log \mathbb{P}_o(u_{i,t}|V_i \setminus u_{i,t}). \tag{5.2.3}$$

Proof.

$$\begin{aligned}
\mathcal{L}_o &= -\sum_{t=1}^{T} \sum_{C_t \in \mathcal{P}_t} \lambda_{C_t} \mathrm{KL}\Big(\widehat{\mathbb{P}}_o(\cdot|C_t), \mathbb{P}_o(\cdot|C_t)\Big) \\
&= -\sum_{t=1}^{T} \sum_{C_t \in \mathcal{P}_t} \sum_{i=1}^{N} \omega_i \mathbf{I}_{\{C_t = V_i \setminus u_{i,t}\}} \cdot \sum_{i=1}^{N} \widehat{\mathbb{P}}_o(u_{i,t}|C_t) \log \frac{\widehat{\mathbb{P}}_o(u_{i,t}|C_t)}{\mathbb{P}_o(u_{i,t}|C_t)} \\
&= -\widehat{C}_o + \sum_{t=1}^{T} \sum_{i=1}^{N} \omega_i \log \mathbb{P}_o(u_{i,t}|V_i \setminus u_{i,t}),
\end{aligned} \tag{5.2.4}$$

where

$$\widehat{C}_o = \sum_{t=1}^{T} \sum_{C_t \in \mathcal{P}_t} \sum_{i=1}^{N} \omega_i \mathbf{I}_{\{C_t = V_i \setminus u_{i,t}\}} \cdot \widehat{\mathbb{P}}_o(u_{i,t}|C_t) \log \widehat{\mathbb{P}}_o(u_{i,t}|C_t)$$

is a constant and the last equation in (5.2.4) follows from the fact that

$$\widehat{\mathbb{P}}_o(u_{i,t}|C_t) = \frac{w_i}{\sum_{i'=1}^{N} \omega_{i'} \mathbf{I}_{\{C_t = V_{i'} \setminus u_{i',t}\}}}.$$

Threfore, we complete the proof. □

Since $\mathbb{P}_o(u_{i,t}|C_{i,t})$ is the probability of observing a subevent of with participating objects V_i with weight ω_i, by Lemma 5.2.1, we have the minimizing the KL divergence is equivalent to maximium likelihood estimation.

5.2.3 Hebe Hyperedge Prediction

Recall that the hyperedge-driven proximity is to predict the hyperedge (through event identifier) given the set of participating objects. Particularly, we formalize the problem as matching the event identifier with

the given set of participating objects. Therefore, we need to estimate the corresponding scoring function between the hyperedge (event identifier) and the set of objects. Desirably, the new scoring function $S(q_i, V_i)$ for event Q_i measures the similarity between the participating object set and the event identifier. The scoring function is defined as follows:

$$S(q_i, V_i) = \langle \mathbf{h}_i, \frac{1}{T} \sum_{t=1}^{T} \mathbf{w}_{v_{i,t}} \rangle, \tag{5.2.5}$$

where the embedding of the event identifier is $\mathbf{h}_i$ and $\mathbf{w}_{v_{i,t}}$ is the embedding for object $v_{i,t} \in V_i^t$. Note that in HEBE-PE the context is the set of participating objects. Therefore, the condition probability of predicting the subevent identifier in the hyperedge is as follows

$$\mathbb{P}_e(q_i|V_i) = \frac{\exp(S(q_i, V_i))}{\sum_{q_j \in \mathcal{Q}} \exp(S(q_j, V_i))}, \tag{5.2.6}$$

where $\mathcal{Q}$ is the set of all event identifiers. Similarly, $\mathbb{P}_e(q_i|V_i)$ corresponds to given V_i selecting q_i from the pool of candidates $\mathcal{Q}$. To preserve the topological structure of events, we minimize the KL divergence between the empirical distribution of $\widehat{\mathbb{P}}_e(\cdot|V)$ and $\mathbb{P}_e(\cdot|V)$, which is equivalent to maximizing

$$\mathcal{L}_e = -\sum_{V \in \mathcal{V}} \lambda'_V \mathrm{KL}\Big(\widehat{\mathbb{P}}_e(\cdot|V), \mathbb{P}_e(\cdot|V)\Big),$$

where V is a set of participating objects, $\mathcal{V}$ is the sample space of V, and λ'_V is the importance of V, $\lambda'_V = \sum_{i=1}^{N} \omega_i \mathbf{I}_{\{V=V_i\}}$. λ'_V is weighted number of hyperedges connecting with participating objects of V. Since we assume there is no duplicated V_i, for subevent q_i, $\lambda'_{V_i} = \omega_i$.

Lemma 5.2.2. *Maximizing $\mathcal{L}_e$ is equivalent to maximizing*

$$L_e = \sum_{i=1}^{N} \omega_i \mathbb{P}_e(q_i|V_i). \tag{5.2.7}$$

Proof.

$$\begin{aligned}
\mathcal{L}_e &= -\sum_{V \in \mathcal{V}} \lambda'_V \mathrm{KL}\Big(\widehat{\mathbb{P}}_e(\cdot|V), \mathbb{P}_e(\cdot|V)\Big) \\
&= -\sum_{V \in \mathcal{V}} \sum_{i=1}^{n} \omega_i \mathbf{I}_{\{V=V_i\}} \sum_{i=1}^{n} \widehat{\mathbb{P}}_e(q_i|V) \log \frac{\widehat{\mathbb{P}}_e(q_i|V)}{\mathbb{P}_e(q_i|V)} \\
&= -\widehat{C}_e + \sum_{V \in \mathcal{V}} \sum_{i=1}^{n} \omega_i \widehat{\mathbb{P}}_e(q_i|V_i) \log \mathbb{P}_e(q_i|V_i),
\end{aligned} \tag{5.2.8}$$

where

$$\widehat{C}_e = \sum_{i=1}^{n} \omega_i \log \widehat{\mathbb{P}}_e(q_i|V_i)\widehat{\mathbb{P}}_e(q_i|V_i)$$

is a constant and the last equation in (5.2.8) follows from the fact that

$$\widehat{\mathbb{P}}_e(q_i|V) = \frac{\omega_i}{\sum_{i'=1}^{N} \omega_{i'} \mathbf{I}_{\{V=V_{i'}\}}}.$$

Threfore, we complete the proof. □

Since $\mathbb{P}_e(q_i|V_i)$ is the probability of observing a subevent of with participating objects V_i in subevent q_i with weight ω_i, by Lemma 5.2.2, we have the minimizing the KL divergence is equivalent to maximium likelihood estimation.

5.2.4 Multiple Event Types

In this subsection, we relax the assumption that there is only one event type in the heterogeneous information network by considering a more general case, where there are multiple event types, such as Example 5.1.3. As depicted in Figure 5.1, there are two event types in the event schema, *i.e.*, business profile event type and review event type. Assume there are K event types, considering HEBE-PO, for each event type, we have the objective function as $\mathcal{L}_o^k$ for $k = 1, \ldots, K$. We treat each event type as equally important. Therefore, the overall objective function to be minimized is:

$$\mathcal{L}_o^* = \sum_{k=1}^{K} \mathcal{L}_o^k.$$

Similar analysis can be directly applied to HEBE-PE.

5.3 Optimization

We introduce a novel general optimization procedure, called *Noise Pairwise Ranking (NPR)*, for the HEBE framework. Then, we apply NPR to both HEBE-PO and HEBE-PE to derive the object embeddings.

5.3.1 Noise Pairwise Ranking

Considering the objective function of HEBE-PO in (5.2.3), direct optimizing $\mathcal{L}_o$ is intractable since the conditional probability (5.2.1) requires the summation over the entire set of objects of type X_1. The same challenge exists for optimizing the objective function of HEBE-PE in (5.2.7), which requires the summation

over the entire set of event identifiers. In the real world applications, the size of objects and event identifiers can be tremendous.

To address this challenge, noise contrastive estimation (NCE) [69] and negative sampling (NEG) [68] are proposed. NCE reduces the problem of estimating the conditional probability into a probabilistic classification problem to distinguish samples from the empirical distribution and a noise distribution, where the empirical distribution corresponds to positive samples and the noise distribution corresponds to negative samples. Moreover, based on NCE, [68] introduces negative sampling. Negative sampling also learns the parameters as a binary classification problem, it particularly formulates the objective as logistic regression, which is shown to be effective in embedding learning [68, 78, 93].

As [31] shows, the hyperparameter of negative sampling value k [68] plays an important role in obtaining the optimal embeddings. To get rid of the hyperparameter, we develop a new optimization framework, called *noise pairwise ranking* (NPR), from a pairwise ranking perspective. In comparison, NCE and NEG are discriminative models, while our model is a generative model in optimizing the conditional probability. The developed NPR framework is applicable to both Hebe-Po and Hebe-Pe. To illustrate the underlying idea, the conditional probability to be maximized can be abstracted as follows:

$$\mathbb{P}(u|C) = \frac{\exp\big(S(u,C)\big)}{\sum_{u' \in \mathcal{U}} \exp\big(S(u',C)\big)}, \tag{5.3.1}$$

where $\mathcal{U}$ is the set of targets (which can be instantiated to objects or event identifiers). For Hebe-Po, $\mathcal{U}$ corresponds to X_t where t is the type of the target object; while for Hebe-Pe, $\mathcal{U}$ corresponds to $\{q_i\}_{i=1}^N$. Therefore, we have

$$\mathbb{P}(u|C) = \Big[1 + \sum_{u' \in \mathcal{U} \setminus u} \exp\big(S(u',C) - S(u,C)\big)\Big]^{-1}, \tag{5.3.2}$$

which follows from (5.3.1) via dividing the denominator and numerator by $\exp\big(S(u,C)\big)$. Instead of directly optimizing (5.3.2) over all $u' \in \mathcal{U} \setminus u$, we sample a sample u_n from $\mathcal{U} \setminus u$ as a negative sample; then we update (5.3.2) using u_n as proxy of $\mathcal{U} \setminus u$. W.r.t. sampling u_n from $\mathcal{U} \setminus u$, similar to NCE and NEG [69, 68], NPR also has a noise distribution P_n as a free parameter.

We use $\sigma(\cdot)$ to denote the sigmoid function that $\sigma(x) = 1/(1+\exp(-x))$. In order to maximize the conditional probability defined in (5.3.1), we maximize the following noise pairwise ranking function [68] instead,

$$\mathbb{P}(u > u_n|C) = \sigma\big(-S(u_n,C) + S(u,C)\big), \tag{5.3.3}$$

which can be interpreted as maximizing the probability of observing the target u over the noise u_n, given

the context C. Particularly, it can be verified as follows that

$$\mathbb{P}(u|C) > \prod\nolimits_{u_n \neq u} \mathbb{P}(u > u_n|C),$$

which implies that optimizing $\mathbb{P}(u > u_n|C)$ can be explained as optimizing the lower bound of $\mathbb{P}(u|C)$.

Remark 5.3.1. *The derived noise pairwise ranking results in (5.3.3) is similar to the Bayesian Pairwise Ranking (BPR) proposed in [81]. However, BPR is designed for the personalized ranking in a specific recommender system with the negative samples coming from missing implicit feedback; while our NPR is derived based on approximation from the softmax definition of the conditional probability, besides the negative samples are sampled from a noise distribution.*

Thus, for all $u_n \in \mathcal{U} \setminus u$, (5.3.1) can be approximated by

$$\log \mathbb{P}(u|C) \propto \mathbb{E}_{u_n \sim P_n} \log \mathbb{P}(u > u_n|C),$$

where P_n is the noise distribution. We set $P_n \propto d_u^{3/4}$, as proposed in [68], where d_u is the degree of u. For HEBE-PO, the degree d_u of each object is the number of hyperedges that object u involves in; while for HEBE-PE, the degree d_u is set to be the weight of the event (i.e., ω_u).

5.3.2 Optimization for Hebe-Po

Based on the NPR optimization framework proposed in Section 5.3.1, we apply it to the method of HEBE-PO in the HEBE framework. Recall that the optimization objective of HEBE-PO is defined in (5.2.3) with the conditional probability defined in (5.2.1). By applying the NPR optimization framework to the conditional probability in (5.2.1), we have the following new objective function

$$\widetilde{\mathcal{L}}_o = -\sum_{i=1}^{N} \omega_i \sum_{t=1}^{T} \mathbb{E}_{u_{n,t} \sim P_n(X_t)} \mathbb{P}_o(u_{i,t} > u_{n,t}|C_{i,t}).$$

where $u_{n,t}$ is the sampled noise from $P_n(X_t)$ and the latter is the noise distribution of objects of type X_t. By applying (5.3.3), we have

$$\mathbb{P}_o(u_{i,t} > u_{n,t}|C_{i,t}) = \sigma\big(-S(u_{n,t}, C_{i,t}) + S(u_{i,t}, C_{i,t})\big).$$

To optimize $\widetilde{\mathcal{L}}_o$, we use the asynchronous stochastic gradient algorithm (ASGD) [80]. ASGD takes advantage of the sparsity of the optimization problem, which means that most gradient updates only modify a small

Algorithm 4 HEBE-PO($Q_i, \boldsymbol{\beta}$)

1: Sample a subevent $\widetilde{Q}$ from Q_i;
2: Uniformly sample an object type X_t from $\widetilde{Q}$;
3: Draw a random object from $P_n(X_t)$ as noise object;
4: Update Object Embeddings $\boldsymbol{\Theta}$ of $\widetilde{Q}$ by Gradient Descent (5.3.4) with type-wise step size $\boldsymbol{\beta}$;
5: **Return:** $\boldsymbol{\Theta}$

portion of the variables. Define $\boldsymbol{\Theta} = \{\mathbf{w}_v\}_{v\in\mathcal{X}}$ as the parameters, where $\mathbf{w}_v$ is the embedding for object v, we have the gradient

$$\frac{\partial \widetilde{\mathcal{L}}_o}{\partial \boldsymbol{\Theta}} = -\sum_{i=1}^{N} \omega_i \sum_{t=1}^{T} \mathbb{E}_{u_{n,t}\sim P_n(\mathcal{X}_t)} \frac{\partial \mathbb{P}_o(u_{i,t} > u_{n,t}|C_{i,t})}{\partial \boldsymbol{\Theta}}.$$

We define the shorthand notation $\mathbb{P}_o(>_{t,i,n}) = \mathbb{P}_o(u_{i,t} > u_{n,t}|C_{i,t})$. The gradients can be written as follows:

$$\begin{aligned}
\frac{\partial \ln \mathbb{P}_o(>_{t,i,n})}{\partial \mathbf{w}_{u_{i,t}}} &= \frac{\sigma(-S_\Delta)\sum_{t=2}^{T}\mathbf{w}_{c_t}}{T-1};\\
\frac{\partial \ln \mathbb{P}_o(>_{t,i,n})}{\partial \mathbf{w}_{u_{n,t}}} &= -\frac{\sigma(-S_\Delta)\sum_{t=2}^{T}\mathbf{w}_{c_t}}{T-1};\\
\frac{\partial \ln \mathbb{P}_o(>_{t,i,n})}{\partial \mathbf{w}_{c_t}} &= \frac{\sigma(-S_\Delta)(\mathbf{w}_{u_{i,t}} - \mathbf{w}_{u_{n,t}})}{T-1}.
\end{aligned} \tag{5.3.4}$$

where $S_\Delta = -S(u_{n,t}, C_{i,t}) + S(u_{i,t}, C_{i,t})$.

Gradient coefficient. Objects in types of smaller sizes are more likely to be sampled when we sample the events. For instance in the example of DBLP, the expected probability of a random venue being sampled is propotional to $1/|X_V|$; while for objects of types author and term, the expected probabilities of being sampled are $1/|X_A|$ and $1/|X_T|$, respectively. Since $|X_A| \gg |X_V|$ and $|X_T| \gg |X_V|$, the expected probability of each venue being sampled is higher than authors and terms. Similar observations can also be made in the Yelp network. This inevitably makes some object types better trained than others as optimization proceeds, resulting in the learned $\boldsymbol{\Theta}$ being trapped at poor local optima during the optimization procedure.

In order to balance the average step size among different object types, when applying ASGD to learn the object embeddings (and event identifier embeddings for HEBE-PE), we propose to adjust the global step size using a type-wise gradient coefficient. Suppose the global step size is η, given an object type t, the step size for each object in X_t is defined as $\beta_t = \alpha_t \eta$, where α_t is the gradient coefficient,

$$\alpha_t = \frac{|X_t|}{\max_{t'=1}^{T}\{|X_{t'}|\}}. \tag{5.3.5}$$

We define $\boldsymbol{\beta} = [\beta_t]_{t=1}^{T}$ as the vector of step size for each object type. By (5.3.5), we have that for object type

Algorithm 5 HEBE-PE($Q_i, \boldsymbol{\beta}$)

1: Sample a subevent $\widetilde{Q}$ from Q_i;
2: Draw an event identifier from $P_n(\mathcal{Q})$ as negative;
3: Update Object Embeddings $\boldsymbol{\Theta}$ of $\widetilde{Q}$ by Gradient Descent (5.3.6) with type-wise step size $\boldsymbol{\beta}$;
4: Update Event Identifier Embeddings $\boldsymbol{H}$ of q_i by Gradient Descent (5.3.7) with type-wise step size $\boldsymbol{\beta}$.
5: **Return:** $\boldsymbol{\Theta}$, $\boldsymbol{H}$

t, the smaller $|X_t|$, the smaller α_t, corresponding to smaller step-size. Therefore, we slow down the training process for the objects of type t where $|X_t|$ is relatively smaller.

The updating process for a single iteration of HEBE-PO is summarized in Algorithm 4.

5.3.3 Optimization for Hebe-Pe

Similarly, we apply the NPR optimization framework to HEBE-PE, which yields the new optimization objective of

$$\widetilde{\mathcal{L}}_e = -\sum_{i=1}^{N} \omega_i \mathbb{E}_{q_n \sim P_n(\mathcal{Q})} \mathbb{P}_e(q_i > q_n | V_i),$$

where q_n is the sampled noise event from $P_n(\mathcal{Q})$. In addition, we have

$$\mathbb{P}_e(q_i > q_n | V_i) = \sigma\big(- S(q_n, V_i) + S(q_i, V_i)\big).$$

It is worth noting that in HEBE-PE, we have event identifier embeddings ($\boldsymbol{H} = \{\mathbf{h}_i\}_{i=1}^N$) as parameters, in addition to object embeddings $\boldsymbol{\Theta}$. The gradient $\partial \mathbb{P}_e(>_{i,n}) / \partial \boldsymbol{\Theta}$ can be obtained as follows, with $\mathbb{P}_e(>_{i,n}) = \mathbb{P}_e(q_i > q_n | V_i)$,

$$\frac{\partial \ln \mathbb{P}_e(>_{i,n})}{\partial \mathbf{w}_{v_{i,t}}} = \frac{\sigma(-S_\Delta)(\mathbf{h}_i - \mathbf{h}_n)}{T}. \tag{5.3.6}$$

where $S_\Delta = -S(q_n, V_i) + S(q_i, V_i)$.

Additionally, we have $\partial \mathbb{P}_e(>_{i,n}) / \partial \boldsymbol{H}$ as follows:

$$\begin{aligned} \frac{\partial \ln \mathbb{P}_e(>_{i,n})}{\partial \mathbf{h}_i} &= \frac{\sigma(-S_\Delta) \sum_{t=1}^{T} \mathbf{w}_{v_{i,t}}}{T}; \\ \frac{\partial \ln \mathbb{P}_e(>_{i,n})}{\partial \mathbf{h}_n} &= -\frac{\sigma(-S_\Delta) \sum_{t=1}^{T} \mathbf{w}_{v_{i,t}}}{T}. \end{aligned} \tag{5.3.7}$$

The corresponding updating process for a single iteration of HEBE-PE is presented in Algorithm 5.

Algorithm 6 HEBE.

1: **Initialize:** randomly initialize $\boldsymbol{\Theta}$, $\boldsymbol{H}$
2: **for** $t = 1, \ldots, T$ **do**
3: Calculate α_t via (5.3.5)
4: **end for**
5: **for** $i = 0$ to $I_N - 1$ **do**
6: $\eta \leftarrow \eta_0 \cdot (I_N - i)/I_N$
7: $\boldsymbol{\beta} \leftarrow \eta \cdot [\alpha_o]_{o \in \mathcal{O}}$
8: **for** $k = 1, \ldots, K$ **do**
9: Sample a event Q_i of event type k
10: **if** method is HEBE-PO **then**
11: $\boldsymbol{\Theta} \leftarrow$ HEBE-PO$(Q_i, \boldsymbol{\beta})$
12: **else if** method is HEBE-PE **then**
13: $\boldsymbol{\Theta}, \mathbf{H} \leftarrow$ HEBE-PE$(Q_i, \boldsymbol{\beta})$
14: **end if**
15: **end for**
16: **end for**
17: **Return:** $\boldsymbol{\Theta}, \mathbf{H}$

5.3.4 Unified Algorithm

The optimization procedures for HEBE-PO and HEBE-PE introduced in the previous sections are applicable when there is only one event type. Here, we consider the more general scenario where there are multiple event types (*i.e.*, $K > 1$). The unified algorithm is described in Algorithm 6, with η_0 and R as the initial step size and the iteration number. When learning embeddings for the objects (and the event identifiers) in the heterogeneous information networks, we opt to use a similar procedure to that used in [92], which is to use all event types jointly and weigh each event type equally. Accordingly, we adopt the strategy that first uniformly samples a event type and then sample a event instance of that type, as shown in Line 8.

5.4 Experimental Study

In this section, we report experimental results of the proposed two HEBE methods, including HEBE-PO and HEBE-PE, corresponding to object-driven proximity and hyperedge-driven proximity, respectively. To evaluate how well the learned embeddings preserve the proximity between objects in heterogeneous information networks with events, we evaluate the embeddings using both classification and ranking measures. Particularly, via a series of quantitative studies, we aim at answering the following questions:

Q1: W.r.t. classification tasks, do HEBE methods, including both HEBE-PO and HEBE-PE, learn better object embeddings compared with existing methods?

Q2: Are HEBE methods robust to random noisy objects included in the event schemata?

Q3: Are HEBE methods robust to data sparseness?

Q3: W.r.t. classification tasks, under what scenarios, does HEBE-PO learn better embeddings than HEBE-PE, and vice versa?

5.4.1 Datasets and Compared Methods

We introduce two datasets on which we conduct experiments: DBLP and Yelp, as of Example 1.3.1 and Example 5.1.3. The basic statistics of both datasets are summarized in Table 5.1. **DBLP** is a collection of bibliographic information on major computer science journals and proceedings, from which we extracted three types of objects and one event type, as shown in Figure 1.2. Each event corresponds to a publication, and each publication involves authors, venue, and terms used in the paper. The **Yelp** dataset provides business reviews and we extracted two event types as presented in Figure 5.1 with review and business profile as their event identifiers. In event type I, there are three object types including user, business and term; while in event type II, we have two object types, business and term used in its name. User type is removed in event type I due to data sparsity that the number of reviews written by each user is typically small. It is worth noting that we distinguish the terms in the review and terms in the business profile in event type II.

Table 5.1: Number of objects for DBLP and Yelp.

DBLP	Author	Term	Venue	Paper
	209,679	165,657	7953	1,938,912
Yelp	Business	Term (review)	Term (name)	Review
	12,241	130,259	6,709	905,658

In order to demonstrate the efficacy of the two proposed methods, we use an extensive set of existing methods as baselines. For the sake of convenience, we define some notations before detailing the baselines. Recall that $\mathcal{X}$ is the set of objects and $\mathcal{D}$ is the set of events. We define the coocurrence matrix $\mathbf{M} \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$ such that $\mathbf{M}_{i,j}$ denotes the number of events that two objects are both involved in. It is worth noting that by constructing the coocurrence matrix, we ignore the type information associated with each object. Due to the fact that some methods decompose the data into pairwise interactions, total degrees among different interactions may vary significantly and compromise the embeddings. For fair comparison, we therefore can first apply degree normalizions to these pairwise interaction sets and then merge them to get normalized coocurrence matrix $\widetilde{\mathbf{M}}$ as presented in [45]. The dimensionality of object embeddings is set to be 300 for all methods. In particular, we consider the following methods:

- Singular Value Decomposition (SVD) on $\mathbf{M}$, and singular vectors are used as object representations.

Table 5.2: Classification accuracy (%) and AUC on two datasets, respecting tasks of research group (DBLP), research area (DBLP) and restaurant categories (Yelp).

	Research Group		Research Area		Restaurant Type	
Method	Acc.	AUC	Acc.	AUC	Acc.	AUC
SVD	81.03	0.7137	83.27	0.5720	74.09	0.7147
NSVD	72.41	0.6958	89.75	0.6271	66.45	0.6244
PPMI	70.69	0.7513	90.22	0.7450	82.82	0.6504
NMF	73.28	0.6210	75.69	0.5798	79.64	0.7955
NNMF	72.41	0.7223	88.31	0.7665	72.00	0.7328
LINE	78.45	0.5607	79.48	0.5565	79.82	0.6378
PTE	**87.93**	0.7235	90.27	0.6646	81.91	0.7195
HEBE-PO	84.48	0.7957	**92.18**	0.7905	**88.00**	**0.8961**
HEBE-PE	87.07	**0.8207**	91.66	**0.8417**	87.27	0.8826

- Normalized SVD (NSVD) on $\widetilde{\mathbf{M}}$.
- Positive shifted PMI (PPMI). As shown in [58], the word embedding with negative sampling is equivalent to approximate the PPMI. Hence, we perform SVD on the PPMI matrix of $\mathbf{M}$. We have $k = 5$ as the negative sampling parameter.
- Non-negative Matrix Factorization (NMF) on $\mathbf{M}$, and matrix factor is used as object representation.
- Normalized NMF (NNMF) on $\widetilde{\mathbf{M}}$.
- LINE [93]: a second-order object embedding approach originally proposed for networked data. We apply LINE to the decomposed pairwise interactions directly, ignoring the object type information.
- PTE [92]: an object embedding approach that applies pairwise modeling in a round-robin fashion within each event, considering the type information.[1]

The implementation of Hebe can be found here (`bitbucket.org/hgui/hebe`).

5.4.2 Evaluation Metric

The goal of our experiments is to quantitatively evaluate how well our methods perform in generating proximity-preserved embeddings.

One way to evaluate the quality of the embeddings is through proximity-related object classification tasks. After obtaining the embeddings of the objects, we feed these embeddings into classifiers including linear SVM and logistic regression to perform classification with five-fold cross validation. Supposing $x \in \mathcal{X}$,

[1] In the original presentation of [92], labels are provided. For fair comparison, we donot provide labels for PTE during training.

Table 5.3: Classification accuracy (%) and AUC on two datasets with extra noisy object types ("year" for DBLP and "zipcode" for Yelp).

	Research Group		Research Area		Restaurant Type	
Method	Acc.	AUC	Acc.	AUC	Acc.	AUC
SVD	78.03	0.6846	80.10	0.5374	67.73	0.6902
NSVD	70.69	0.6668	87.48	0.6112	48.81	0.6138
PPMI	68.09	0.7175	88.99	0.7162	81.09	0.6892
NMF	72.73	0.6121	71.96	0.5635	67.00	0.7469
NNMF	71.38	0.6823	86.12	0.7411	43.45	0.6142
LINE	80.17	0.5465	78.94	0.5425	76.09	0.6035
PTE	85.34	0.6297	89.83	0.5873	75.18	0.6702
HEBE-PO	76.72	0.7582	89.11	0.7614	85.91	0.8296
HEBE-PE	**85.34**	**0.8214**	**91.26**	**0.8425**	**86.73**	**0.8834**

we define l_x^* as the true label of x while $\widehat{l}_x$ as the predicted label of x. We report the classification metric accuracy (Acc.).

$$\text{Acc.} = \frac{1}{|X_l|}\sum\nolimits_{x \in X_l} \delta\big(\widehat{l}_x = l_x^*\big),$$

where X_l is the set of objects that have labels and $\delta(\cdot)$ is the indicator function. Due to the space limit, the higher accuracy of linear svm and logistic regression for each method gets reported.

Classification relies on ground truth labels to learn mapping function between embeddings and classes. It may not be able to exploit information underlying all dimensions. For instance, some dimensions may be independent of the class labels. Therefore we further use a ranking metric called area under the curve (AUC) [23] to evaluate the quality of embeddings over all dimensions.

$$\text{AUC} = \frac{1}{|X_l|}\mathbb{P}\big(\text{sim}(u,v) > \text{sim}(u',v) | l_v^* = l_u^*, l_v^* \neq l_{u'}^*\big),$$

where $v, u, u' \in X_l$ and $\text{sim}(u,v)$ is the similarity measure between the embeddings of objects u, v. Specifically, we use cosine similarity as the similarity measure [68]. The AUC measure becomes high if embeddings are close for objects sharing the same label, and distant for objects having different labels.

Regarding the DBLP dataset, we have two types of labels of authors. The first is on the **research groups**, with 116 members from four research group manually labelled. These groups are lead by Christos Faloutsos, Dan Roth, Jiawei Han, and Michael I. Jordan, respectively. The other type of labels is on the **research area**, including 4,040 researchers from four research areas including data mining, database, machine learning, and artificial intelligence.

As for the Yelp dataset, we select eleven **restaurant categories** including Mexican, Chinese, Italian, American (traditional), American (new), Mediterranean, Thai, French, Japanese, Vietnamese and Indian as

labels. [2] For each category, we randomly select 100 restaurants that have at least 50 reviews. Restaurants with multiple labels are excluded.

5.4.3 Experimental Results

Now we are ready to present the experimental results for the aforementioned tasks and try to answer the three questions raised at the beginning of this section.

Classification Results

Table 5.2 summarizes the experimental results on classification (Acc.) and ranking (AUC) in DBLP and Yelp.

Considering the results for research group classification in DBLP, we note that PTE and HEBE-PE achieve the best performance. PTE is slightly better than HEBE-PE on accuracy but the latter outperforms the former on AUC by a large margin. HEBE-PO narrowly loses to HEBE-PE on both measures. It is interesting to see that the normalization strategy on $\mathbf{M}$ has a big effect on the performance, but the trend is oppsite between SVD and NMF respecting AUC.

For the task of research area classification in DBLP, HEBE-PO attains the best performance on classification accuracy and HEBE-PE has the highest AUC score. The results on research area are better than the ones on research group for all methods, which means that the research area task is easier than the former task. It's worth noting that two HEBE methods are better than baselines on both measures, confirming the their effectiveness of capturing the proximity. We also observe that both NSVD and NNMF beat their unnormalized versions, implying that the normalization trick works at least for some tasks.

With respect to the Yelp dataset, on classifying the restaurant type, we observe that both HEBE methods are significantly better than the baselines, for both measures. A tentative explanation is that HEBE framework models both event types explicitly, the review event and the business profile event, which better captures the proximity among objects. For PTE and the rest methods, this intricate structure will be dropped due to the representation limits of the models.

To summarize, we positively answer Q1 on the effectiveness of HEBE methods in learning the object embeddings. Among all the competitors, PTE works relatively well for all three tasks, showing its idea of modeling pairwise interactions better than the rest. But compared to our framework, by modeling each event as a whole, one can achieve even better performance.

[2] The labels are obtained from www.yelp.com.

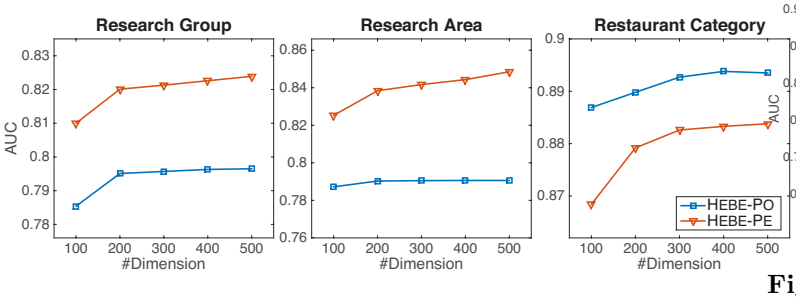


Figure 5.3: Performance variations in terms of AUC verse the dimension of the embeddings.

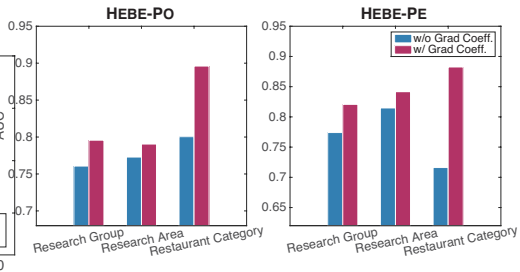


Figure 5.4: Performance variations in terms of AUC verse the choice of the gradient for updates.

Robustness to Noisy Objects

One challenge of modelling events in heterogeneous information networks is to develop a method with anti-interference ability. Hence, we test the robustness of the HEBE framework against artificially inserted object noises. The added noisy objects are designed to convey little knowledge regarding the tasks on both datasets. Consequently, for DBLP data, we include the year of the publication as an additional object type. For Yelp data, the zip code of each restaurant is considered. The results are summarized in Table 5.3.

For all three tasks, HEBE-PE achieves the best performance and is better than the baselines by a large margin. In addition, HEBE-PO is bested by HEBE-PE for all three tasks, but attain results better than PTE and the rest methods in most cases. These observations verify our expectation that HEBE-PE is more robust to noise than all the rest methods including HEBE-PO. A possible explanation is that HEBE-PO explicitly models the proximity between the noisy object and the context objects, leading to deviation of the object embeddings from the optimal ones. In contrast, HEBE-PE additionally learns the event identifier embeddings. With respect to each event, the event identifier serves as a filter and summarizing the sematic information of all participating objects. Since the noise objects have low semantic similarity with the other participating objects, information related to the noise objects will be droped by the event identifier embedding. While learning embeddings for other objects, information is directly propagated from the event identifier to objects. Consequently, the object embedding learning will not be influenced by the noise objects. The experimental results across all three tasks, we have HEBE-PO achieves the best performance. Moreover, we can observe that by adding noise, HEBE-PO achieves good classification accuracy as when there are no noise objects. On the other hand, we still recognize that HEBE-PO is the second best method in terms of absolute performance.

Robustness to Sparsity

In general, the sparsity of event data is defined as the average number of events each object is involved in. Thus, if we assume the set of objects to be relatively stable, the sparsity of the heterogenous event data can be

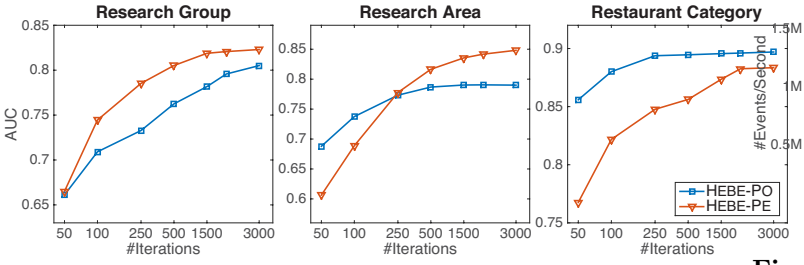


Figure 5.5: Performance variations in terms of AUC verse the number of updating iterations.

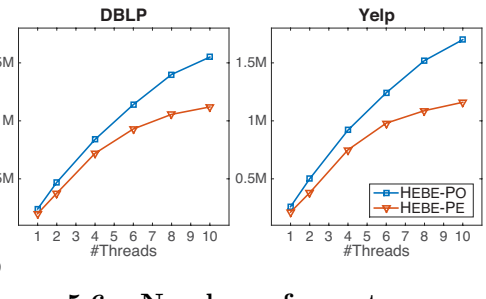


Figure 5.6: Number of events processed per second verse the number of threads.

altered by sampling a subset of all events. In this section, we randomly sample different percentages (1%, 5%, 10%, 20%, 30%, 50%) of the two datasets and repeat the three tasks mentioned aforehand. Experimental results are reported in Table 5.4 for the DBLP dataset and Table 5.5 for the Yelp dataset. The density measures are reported in the first two rows. For DBLP, since the classification is performed on authors, we define **density measure** as the number of publications each author is associated with. For Yelp, because the businesses are of interest, we define **density measure** as the number of reviews each restaurant receives. The density measure increases as the sampling percentage increases, and its incremental rate is slower than the latter due to the long-tail behavior in the event data. In other words, when more events are sampled, the size of the object set will also increase, but having a slower rate of increment.

Across the three tasks in the two datasets, vertically we observe the two HEBE methods achieve the best performance in general among all cases. In DBLP dataset, for both tasks, HEBE-PE is better than HEBE-PO for both measures in most cases. In Yelp dataset, when less than 20% of events are sampled, HEBE-PE attains better results than HEBE-PO; when more than 20% of events are sampled, HEBE-PO outperforms similar to HEBE-PE; across the different sampling percentages the margin between HEBE-PO and HEBE-PE is relatively small. For different percentages, we observe that PTE is still the most stable method among all baselines while the performances of the rest fluctuate wildly for different tasks. When the density measure is close to 1 such as 1% of events being sampled in the DBLP dataset, the AUC scores are close to random (0.5). This is because with a density measure of 1.29, the average number of events an object is involved in is only slightly higher than 1 and the co-occurrence observations are not sufficient to capture proximity among objects.

Based on the vertical comparison from Table 5.3, with regard to Q2, the HEBE framework is relatively robust to noise and data sparsity. For the scenarios (i) when there is noise objects and (ii) when the observation data is sparse, HEBE consistently outperforms the baselines.

Horizontally, we observe that when more events are observed, the accuracies of the classification tasks in-

creases as well.The increment rate is the largest when sampling percentage changes from 1% to 5%. Similarly, the performance improvements from 10% to 20% are more significant than from 20% to 30%. Particularly, we are interested in the case, when the sampled percentage of events exceeds 20%, the performance of HEBE-PO becomes comparable with HEBE-PE, and is even slightly better. When the density measure increases, HEBE-PO becomes more effective in modeling the semantic relatedness. In other words, HEBE-PO is more effective when there are enough observations. It is worth noting that even though HEBE-PO better performs than HEBE-PE when the sampling percentage is larger than 20%, HEBE-PE is only surpassed by a small margin.

Hence, we answer Q3 based under two scenarios. If the data is noisy, HEBE-PE is more robust than HEBE-PO. If the data is relatively sparse, HEBE-PE is more effective than HEBE-PO; otherwise, if the data is relatively dense, both methods are robust in preserving the proximity among objects.

5.4.4 Model Study

In this section, we study the effect of the hyperparameters. Particularly we study four aspects: the dimensionality of the embedding, the type-wise gradient coefficient, the number of iterations, and the number of threads. Based on studies in Section 5.4.3, we observe that AUC results are more stable than classification accuracy. Our explanation is that classification needs to learn the mapping function between embeddings and classes based on some certain assumptions, which may not agree with the embedding data. Therefore, we opt to report AUC results for the model study.

We plot the AUC results against dimensionality of the learned embeddings in Figure 5.3. An increasing and converging performance pattern is observed for both methods, which is a common pattern that has been observed in previous work [93, 92].

In Section 5.3, we proposed a type-wise gradient coefficient for ASGD. We verify the effectiveness of the proposed gradient coefficient, compared with a global gradient for all types, the results are reported in Figure 5.4, which clearly shows the superiority of the proposed gradient coefficient for step size adjustment, especially on the Yelp dataset.

In addition, we study how the number of iterations affects the results, as reported in Figure 5.5. With Y-axis as AUC, the pattern of first increasing and then converging is observed. When the iteration number is sufficiently large, the embeddings are stable.

Regarding the efficiency, we have tested the number of events processed per second against the number of threads, which is shown in Figure 5.6. The experiments were conducted on a machine with 2 Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz (20 Cores). One can observe that the more threads we have, the

larger the number of events processed per second. Therefore, our method can be easily scaled to extremely large information networks. However, it is worth mentioning that the incremental speed-up of HEBE-PE is smaller than HEBE-PO. Our explanation is that HEBE-PE has many more parameters than HEBE-PO due to the embeddings of hyperedge, resulting in slower performance due to the caching mechanism among different threads when they are accessing random objects and hypeedges.

Table 5.4: The classification accuracy and AUC results on sampled DBLP data considering both research group and research area classification. The sparsity is measured by the average number of publication each author is involved in (similar below).

Sampling %.	1%		5%		10%		20%		30%		50%	
Density Measure	1.264		2.028		2.882		4.595		6.400		10.315	
Method	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
Research Group												
SVD	38.46	0.5602	66.67	0.6169	65.59	0.6481	72.55	0.6494	72.86	0.6720	77.28	0.6924
NSVD	43.59	0.5504	58.73	0.5919	68.82	0.6330	70.59	0.6345	72.64	0.6517	74.55	0.6790
PPMI	46.15	0.5502	60.32	0.5993	76.34	0.6557	71.57	0.6703	72.97	0.6792	74.55	0.7192
NMF	41.03	0.5583	57.14	0.5989	56.99	0.5874	54.90	0.6009	66.96	0.5950	70.91	0.6120
NNMF	46.15	0.5462	55.56	0.6601	60.22	0.6806	75.49	0.7167	70.55	0.7197	71.82	0.7294
LINE	**56.41**	0.6004	66.67	0.6254	72.04	0.5877	77.45	0.5619	77.86	0.5669	85.45	0.5871
PTE	**56.41**	0.6190	69.84	0.6727	76.34	0.6434	84.31	0.6778	**85.94**	0.7034	**88.18**	0.6783
HEBE-PO	53.85	0.6034	66.67	0.7082	72.04	0.7151	74.51	0.7515	75.55	0.7640	82.73	0.7841
HEBE-PE	**56.41**	**0.6547**	**73.02**	**0.7434**	**83.87**	**0.7749**	**85.29**	**0.8221**	84.13	**0.8220**	**88.18**	**0.8316**
Research Area												
SVD	47.88	0.5162	62.47	0.5337	66.27	0.5411	71.66	0.5516	75.47	0.5551	79.15	0.5644
NSVD	52.39	0.5076	66.21	0.5004	72.15	0.5021	77.91	0.5157	80.13	0.5299	85.23	0.5600
PPMI	51.67	0.5063	68.00	0.5092	72.66	0.5180	78.15	0.5395	80.59	0.5669	85.91	0.6203
NMF	43.37	0.5143	53.54	0.5329	59.30	0.5391	63.63	0.5493	68.01	0.5560	70.72	0.5637
NNMF	50.50	0.5303	62.50	0.5773	67.73	0.6206	72.37	0.6486	76.51	0.6807	82.91	0.7594
LINE	57.17	0.5552	69.83	0.5764	72.15	0.5716	74.89	0.5501	74.53	0.5339	80.82	0.5822
PTE	53.29	0.5291	**71.54**	0.5858	73.95	0.5782	79.03	0.6015	82.68	0.6356	86.80	0.6340
HEBE-PO	**57.53**	**0.5635**	69.71	0.6108	74.91	0.6798	80.26	0.7199	81.66	0.7293	86.17	0.7817
HEBE-PE	54.64	0.5500	71.09	**0.6282**	**75.90**	**0.6834**	**81.64**	**0.7405**	**83.94**	**0.7645**	**87.84**	**0.8075**

Table 5.5: The classification accuracy and AUC results on sampled Yelp data.

Sampling %.	1%		5%		10%		20%		30%		50%	
Density Measure	1.963		4.791		8.155		15.09		22.32		37.01	
Method	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
SVD	64.12	0.6133	70.85	0.6786	73.44	0.7001	73.98	0.7100	73.82	0.7121	74.82	0.7134
NSVD	62.07	0.6081	63.36	0.6236	65.17	0.6308	66.97	0.6275	67.00	0.6280	67.36	0.6259
PPMI	59.35	0.5561	65.01	0.5484	69.94	0.5626	75.43	0.5824	78.55	0.6089	80.55	0.6253
NMF	63.61	0.6790	71.23	0.7381	75.09	0.7594	76.34	0.7877	78.09	0.7907	78.18	0.7991
NNMF	60.71	0.6710	66.76	0.7022	68.47	0.7082	70.79	0.7213	70.73	0.7297	70.73	0.7312
LINE	60.88	0.5337	71.72	0.5367	77.32	0.5689	80.71	0.6665	80.91	0.6789	81.27	0.6833
PTE	64.29	0.6315	72.89	0.6758	76.28	0.6993	79.25	0.7163	81.00	0.7043	80.91	0.7266
HEBE-PO	71.09	0.7576	79.01	0.8316	82.63	0.8621	85.08	**0.8825**	**86.36**	**0.8845**	**86.82**	**0.8938**
HEBE-PE	**73.30**	**0.7747**	**79.69**	**0.8434**	**83.06**	**0.8746**	**85.44**	0.8779	85.82	0.8765	86.36	0.8862

Chapter 6

Locally-trained Embedding Learning For Expert Finding

6.1 Preliminary

In this section, we include preliminary and problem definition for locally-trained embedding learning for expert finding.

6.1.1 Heterogeneous Bibliographical Networks

A heterogeneous bibliographical network is constructed from bibliographical data. Due to the heterogeneity of the object types, a heterogeneous bibliographical network is naturally a heterogeneous information network [90]. The formal definition of heterogeneous information networks can be found as follows.

Definition 6.1.1 (Heterogeneous Information Network)**.** *For an information network $G = (\mathcal{V}, \mathcal{E})$ with an object mapping function $\phi : \mathcal{V} \to \mathcal{A}$ and an edge mapping function $\psi : \mathcal{E} \to \mathcal{R}$, where $\mathcal{A}$ and $\mathcal{R}$ are the set of object types and edge types, respectively, if the number of object types $|\mathcal{A}| > 1$ or the number of edge types $|\mathcal{R}| > 1$, G is a heterogeneous information network; otherwise, it is a homogeneous information network.*

DBLP [1] is a public bibliographical dataset in the Computer Science domain. We further extract semantic phrases from the text data following the method proposed by Liu et al. [63]. Therefore, we use terms to refer both words and phrases in the corpus. Regarding each publication entry, DBLP provides detailed information about *authors*, *terms*, *venues*. Figure 1.4(a) depicts the network schema and Figure 1.4(b) is a sub-network with a user query. We define the set of publications as $\mathcal{D}$, authors as $\mathcal{A}$, terms as $\mathcal{T}$, and venues as $\mathcal{V}$, with N_D, N_A, N_T, N_V denoting the set sizes, accordingly.

6.1.2 The Document-based Models

The problem of expert finding has been studied extensively [3, 25, 29, 101, 123]. For completeness, we present probably the most popular method: document-based models. The family of document-based models

[1] dblp.uni-trier.de

formalizes the problem as a retrieval task. Given a query q, the ranking score of a researcher candidate a can be calculated as

$$s_c(a, q) \propto \sum\nolimits_{d \in \mathcal{D}} \mathbb{P}(a|d)\mathbb{P}(q|d)\mathbb{P}(d), \tag{6.1.1}$$

where $\mathcal{D}$ is the document corpus, $\mathbb{P}(a|d)$ is the probability that the candidate a is relevant to the publication d, $\mathbb{P}(q|d)$ is the probability that the query q is relevant to the document d, and $\mathbb{P}(d)$ denotes the preference over d.

What remains is to estimate $\mathbb{P}(d), \mathbb{P}(q|d)$, and $\mathbb{P}(a|d)$. Following the ideas of Deng et al. [26], we estimate $\mathbb{P}(p)$ via $\mathbb{P}(d) \propto \ln(e + c_d)$, where c_d is the count of citations of d and e is the mathematical constant to guarantee that weight factor is no less than one. $\mathbb{P}(a|d)$ is generally estimated as $1/|\mathcal{A}_d|$, with $\mathcal{A}_d$ as the set of authors for publication d. Finally, $\mathbb{P}(q|p)$ is calculated based on the query generation retrieval method with Dirichlet prior smoothing [118],

$$\mathbb{P}(q|p) \propto \exp\Big(\sum\nolimits_{t \in q} \mathbb{P}(t|q) \log \mathbb{P}(t|\theta_p)\Big), \tag{6.1.2}$$

where $\mathbb{P}(t|q) = \#(t, q)/\#(q)$ with $\#(t, q)$ as term frequency of term t in q and $\#(q)$ as the length of q and $\mathbb{P}(t|\theta_p)$ is defined as

$$\mathbb{P}(t|\theta_p) = \beta\mathbb{P}(t|p) + (1 - \beta)\mathbb{P}_{\mathrm{b}}(t), \tag{6.1.3}$$

with $\beta = 0.5$ and $\mathbb{P}_{\mathrm{b}}(t)$ as the background language model of the text corpus $\mathcal{D}$.

6.1.3 Word Embedding Learning

Word embedding learning [68, 77] is to represent the terms in a corpus into a low-dimensional latent semantic space, where each term is represented via a low-dimensional vector, which is called embedding or distributed representation. The semantic information regarding each term is preserved such that terms with similar semantic meanings are close to each other in the Euclidean space. There are many off-the-shelf embedding learning algorithms. We adopt word2vec [68] to learn the embeddings, and other embedding methods, such as Latent Semantic Indexing and Glove [77], can also be applied to. In word2vec, for a pair of words that co-occur in a sliding window, one term u is denoted as target and the other v as context. Based on the

skip-gram model, the conditional probability of observing u given c is defined using the softmax function

$$\mathbb{P}(u|c) = \frac{\exp(\mathbf{v}_u^\top \widetilde{\mathbf{v}}_c)}{\sum_{u' \in \mathcal{T}} \exp(\mathbf{u}_{u'}^\top \widetilde{\mathbf{v}}_c)}, \tag{6.1.4}$$

where $\mathbf{v}_u, \widetilde{\mathbf{v}}_c \in \mathbb{R}^z$ are the embeddings for u and c, with z as the dimension of the embedding vector. In (6.1.4), since the denominator sums over all the terms in the corpus $\mathcal{T}$, it is computationally intractable. Consequently, negative sampling is proposed [68]. For the term pair of (u, c), regarding (6.1.4), the following objective is optimized instead,

$$\ell(u, c) = \log \sigma(\mathbf{v}_u^\top \widetilde{\mathbf{v}}_c) + \sum\nolimits_{i=1}^{g} \mathbb{E}_{u_n \sim P_n} \big[\log \sigma(-\mathbf{v}_{u_n}^\top \widetilde{\mathbf{v}}_c) \big], \tag{6.1.5}$$

where $\sigma(\cdot)$ is the sigmoid function, $\mathbf{v}_{u_n} \in \mathbb{R}^d$ is the embeddings of noise u_n, P_n is the noise term distribution, and g is the negative sampling parameter. Due to space limit, one may refer to word2vec [68] for technical details.

6.2 Local Embedding via Concept Hierarchy

Word embedding learning is proposed for *global* embedding learning such that an embedding vector is learned for each term regarding the whole corpus. According to Levy et al. [58], the word embedding learning with negative sampling in (6.1.5) can be loosely interpreted as an implicit matrix factorization problem, where the shifted positive Pointwise Mutual Information (PMI) matrix is approximated by a low-rank matrix with rank equivalent to z (the dimension of the vector space). However, such an approximation may lead to coarse representations of specific terms. The term "information extraction" is not only close to "information extraction" and "named entity recognition" but also to "text mining" and "natural language processing". Suppose "natural language processing" was used as expansion of "information extraction", there will be a semantic drift. Instead of obtaining experts on "information extraction" only, we may also find experts on "natural language processing". However, not all of the experts on "natural language processing" are working on "information extraction".

6.2.1 Concept Hierarchy

In order to address the semantic drift, we relax the global low-rank assumption and propose to represent the terms in the corpus using locally-trained embeddings. In particular, we make the following assumption.

Assumption 6.2.1. *The shifted positive PMI matrix is low-rank for a sub-corpus that is relevant to the*

Table 6.1: List of terms most similar to "Information Extraction" by different embedding methods: (i) Global embedding; (2) the method proposed by Diaz et al. [27] without a concept hierarchy (LE wo/ CH); (3) the proposed locally-trained embedding learning with a concept hierarchy as guidance (LE w/ CH).

Global Embedding	LE wo/ CH [27]	LE w/ CH
information-extraction-ie	pattern-discovery	information-extraction-ie
text-mining	knowledge-based	SystemT [2]
natural-language-processing	indices	ontology-based-information-extraction
question-answering	legal	web-information-extraction
named-entity-recognition	turkish	relation-extraction
nlp	offer	named-entity-recognition

query.

The sub-corpus is constructed with guidance from a concept hierarchy (Figure 1.3(b)). In other words, instead of learning embeddings to preserve the information in the whole corpus, we only preserve information in the sub-corpus. The sub-corpus corresponds to the cluster that "information extraction" belongs to. According to Figure 1.3(b), "information extraction" belongs to the cluster of "natural language processing". Therefore, the sub-corpus comprises publication documents constrained on "natural language processing".

Why using a concept hierarchy as guidance? Regarding the task of expert finding, for a given query "information extraction", the (implicit) background information is "natural language processing". By taking advantage of concept hierarchy, we can identify the background information, as shown in Figure 1.3(b). Alternatively, without a concept hierarchy, as proposed by Diaz et al. [27], the sub-corpus is constructed by retrieving all the documents relevant to the "information extraction". The results obtained following the idea of Diaz et al. [27] are shown in the second column of Table 6.1. However, the top-ranked terms are random and irrelevant to "information extraction". This is because when learning term embeddings on sub-corpus constrained on "information extraction", the term "information extraction" becomes the background since it appears in almost all the documents and (almost) co-occur with all words in the corpus, especially for short documents. In the bibliographical data that we use, around 76% of the document entries are titles. Therefore, "information extraction" is similar to stop words. Meanwhile, if the sub-corpus is constrained on "natural language processing", the term "natural language processing" becomes the background and is distant from "information extraction", as shown in the third column of Table 6.1.

6.2.2 Locally-trained Embedding Learning

How to use concept hierarchy as guidance for local embedding learning? For brevity, we first consider the case where there is only one term in each query, corresponding to one concept in the concept hierarchy. Also, we assume that terms in the query can be trivially mapped to the concept hierarchy. For queries with

more than one concepts, we train local embeddings one by one. For each concept, we use the learned local embeddings to expand the concept accordingly.

For a given query q in the concept hierarchy, we denote the path from root to q as $\mathcal{C}_0 \to \mathcal{C}_1 \to \ldots \to \mathcal{C}_l = q$, where l is the level of the concept hierarchy that q lies at and $\mathcal{C}_0$ corresponds to the root. We use $\{\mathbf{v}_t^m\}_{t\in\mathcal{T}}$ for $m = 0, \ldots, l$ to denote the learned embeddings for terms at level m. The idea of local embedding learning is to *find the nearest neighbors (i.e., expansions) of* $\mathcal{C}_m$ *based on the term embeddings learned constrained on* $\mathcal{C}_{m-1}$. Therefore, the nearest neighbors of q can be found based on the embeddings learned on a sub-corpus constrained on $\mathcal{C}_{l-1}$. In the following, we use "information extraction" as a running example.

For the (sub-)corpus constrained on concept $\mathcal{C}_0$, it is straightforward that we use the whole corpus to train terms' embeddings (i.e., global embeddings). For the corpus constrained on concept $\mathcal{C}_m$ for $m = 1, 2 \ldots$, we first search for the k nearest neighbors of $\mathcal{C}_m$, which serve as expansions to close the vocabulary gap while constructing the sub-corpus. For the query "information extraction", we have $\mathcal{C}_1$ = "natural language processing".

Due to the fact that we do not have features for each concept (and term), we use the embeddings learned via a sub-corpus constrained on concept $\mathcal{C}_{m-1}$ as features. Given $\{\mathbf{v}_t^{m-1}\}_{t\in\mathcal{T}}$ as the embedding learned constrained on $\mathcal{C}_{m-1}$, we use cosine similarity to measure the similarity between term $s_{m-1}(t_1, t_2)$. The top k terms measured by $s_{m-1}(\cdot, \mathcal{C}_m)$ is denoted $\mathcal{N}_m$, as expansion of concept $\mathcal{C}_m$. Therefore, a sub-corpus constrained on $\mathcal{C}_m$ can be extracted based on $\mathcal{N}_m$. In other words, we use global embeddings ($\{\mathbf{v}_t^0\}_{t\in\mathcal{T}}$) to firstly find the query expansions of "natural language processing", which is denoted as $\mathcal{N}_1$. $\mathcal{N}_1$ = { "natural language processing", "nlp", "natural language understanding", "language processing",...}. We interpolate such semantic similarity into the language model with parameter $\gamma \in [0, 1]$,

$$\mathbb{P}^m(t|d) = \gamma\mathbb{P}(t|d) + (1-\gamma)s_{m-1}(t, \mathcal{C}_m)\mathbf{I}(t \in \mathcal{N}_m), \tag{6.2.1}$$

where $\mathbf{I}(w \in \mathcal{N}_m)$ is an indicator function. Substituting (6.2.1) into (6.1.3) and (6.1.2), we obtain $\mathbb{P}^m(q|d)$:

$$\mathbb{P}^m(q|d) = \mathbb{P}^m(t|d) = \beta\mathbb{P}^m(t|d) + (1-\beta)\mathbb{P}_b(t), \tag{6.2.2}$$

where query $q = \{t\}$ contains only one concept, t. In order to train local embeddings on the sub-corpus constrained on $\mathcal{C}^{m-1}$, we sample each document with probability proportional to $\mathbb{P}^{m-1}(q|d)$. We set $\mathbb{P}^0(q|d) = 1/|\mathcal{D}|$, as the uniform sampling. While applying word2vec for embedding learning, in order to estimate the empirical distribution of terms in the sub-corpus constrained on $\mathcal{C}_m$, the sampling weights of each document (i.e., $\mathbb{P}^m(q|d)$) should be considered. The recursive embedding learning framework is detailed

Algorithm 7 Local Embedding Learning via Concept Hierarchy.

Input: Document corpus $\mathcal{D}$, the path to query as $\mathcal{C}_0 \to \mathcal{C}_1 \to \ldots \to \mathcal{C}_l = q$.
Initialize: $\mathcal{S} = \mathcal{D}$.
for $m = 0, \ldots, l-1$ **do**
  Learn embeddings of $t \in \mathcal{T}$ using word2vec
    Sample each document with probability $\propto \mathbb{P}^m(q|d)$
  Output $\mathbf{v}_t^m$ as the embeddings of term t
  Compute $\mathbb{P}^{m+1}(q|d)$ according to (6.2.2).
end for
Return: $\mathbf{v}_t^m$.

in Algorithm 7.

6.3 Expert Ranking in Relevance Network

In order to rank researcher candidates for each query, we have two key insights. (i) A candidate may have papers on many topics. For a given query, only the relevant papers can serve as textual evidence for expertise; (ii) Citation may have time-delay factor. Papers which are published in a highly-ranked venue are likely to be important. Therefore, venues play an important role for ranking.

6.3.1 Relevance Network Construction

For a given query q, we first retrieve all the relevant documents, the set of which is denoted as $\mathcal{D}(q)$; that is $\mathcal{D}(q) = \left\{d : \prod_{t_i \in q} \max_{t' \in \widetilde{\mathcal{N}}_l(t_i)} \left\{\mathbf{I}(t' \in d)\right\} = 1\right\}$, where $\mathbf{I}(t' \in d) = 1$ if t' is within the document d, 0 otherwise. In other words, we select all the papers that contain at least one relevant term in $\widetilde{\mathcal{N}}_l(t_i)$ for each term t_i $(i = 1, 2, \ldots, N_q)$ in q.

Based on $\mathcal{D}(q)$, a relevance sub-network can be extracted from the heterogeneous bibliographical network by extracting $\mathcal{D}(q)$ and associated authors and venues. LE-expert ranks the candidates within the relevance sub-network.

6.3.2 Ranking in Relevance Network

To rank candidates for each query, we take advantage of the network structure and propose a ranking algorithm to estimate the authority of objects in the sub-network based on a coupled random walk in the relevance sub-network. We first present the ranking method in a general framework, which can be generalized for other heterogeneous information networks.

Suppose there are M types of objects in the heterogeneous information network and the set of the type i objects is denoted as $\mathcal{V}_i$. The network is represented by a set of relation matrices $\mathcal{R} = \{\mathbf{R}_{ij}\}_{i,j=1}^{M}$. For each

$\mathbf{R}_{ij}$, we define a diagonal matrix $\mathbf{D}_{ii}$ such that the diagonal element at (a,a) of $\mathbf{D}_{ij}$ is the sum of a-th row of $\mathbf{R}_{ij}$. Therefore, the transition matrix of $\mathbf{R}_{ij}$ is defined as $\mathbf{P}_{ij} = \mathbf{D}_{ij}^{-1}\mathbf{R}_{ij}$. For the ranking score vector of objects in type i can be updated iteratively:

$$\mathbf{r}_i^t \propto \frac{\sum_{j=1}^{M} \lambda_{ji}\mathbf{r}_j^{t-1}\mathbf{P}_{ji} + \eta_i \mathbf{r}_i^0}{\sum_{j=1}^{M}\lambda_{ji} + \eta_i}, \tag{6.3.1}$$

where t is the iteration step and $\mathbf{r}_i^0 = \mathbf{1}/|\mathcal{V}_i|$. The relative importance of neighbors of different types is controlled by $\lambda_{ij} \in [0,1]$.

Regarding the task of expert finding in heterogeneous bibliographical networks, the random rank is designed regarding the following assumption.

Assumption 6.3.1. *(a) High-quality and relevant papers will be frequently cited by many other relevant papers;*

(b) Relevant highly-ranked experts will publish many high-quality and relevant papers, and vice versa.

(c) Relevant and highly-ranked conferences attract many high-quality and relevant papers, and vice versa.

Concretely, for the task of expert ranking in the relevance network, there are the following relations for each object type. **Paper:** (i) Citation relations. $\mathbf{R}_{\mathrm{PP}}(a,b) = 1$ if the paper a cites the paper b; (ii) Write relations. $\mathbf{R}_{\mathrm{AP}}(a,b) = 1$ if a-th author writes b-th paper; (iii) Publish relations. $\mathbf{R}_{\mathrm{VP}}(a,b) = 1$ if a-th author writes b-th paper. **Author:** (i) Coauthor relations. $\mathbf{R}_{\mathrm{AA}} = \mathbf{R}_{\mathrm{AP}}\mathbf{R}_{\mathrm{AP}}^{\top}$; (ii) Write^{-1} relations. $\mathbf{R}_{\mathrm{PA}} = \mathbf{R}_{\mathrm{AP}}^{\top}$. **Venue:** (i) Citation relations. $\mathbf{R}_{\mathrm{VV}} = \mathbf{R}_{\mathrm{VP}}\mathbf{R}_{\mathrm{PP}}\mathbf{R}_{\mathrm{VP}}^{\top}$ (ii) Publish^{-1} relations. $\mathbf{R}_{\mathrm{PV}} = \mathbf{R}_{\mathrm{VP}}^{\top}$.

Since terms are used to construct the relevance network and terms do not reflect authority, we do not consider terms while ranking the candidates.

Remark 6.3.2. *The underlying philosophy of the ranking module is similar to NetClus [90] and RankClass [45]. However, NetClus and RankClass are primarily designed for clustering and classification in the whole heterogeneous information network, respectively; while LE-expert is designed for authority ranking within a relevance sub-network. In addition, NetClus can only be applied to star-schema heterogeneous networks while LE-expert is independent of the network schema. Moreover, RankClass is a regularization framework for label propagation whereas LE-expert is based on random walks.*

6.4 Experimental Results

In this section, we present the experimental results as support for the effectiveness of the proposed framework.

6.4.1 Experimental Setup

Data. To evaluate the proposed framework, we conduct numerical experiments and case studies on the dataset of DBLP. In the DBLP dataset, there are 2,244,018 papers, 1,274,360 authors, 8,882 venues, and 1,812,277 words and phrases. Among all the papers, 529,498 papers (24%) have abstract information. The labelled dataset is from Deng et al. [25], which contains 20 queries in total, including both general and specific ones. Details on the queries and the number of experts for each query can be found therein [25].

Evaluation Metrics. Regarding evaluation of the task, we employ several popular information retrieval metrics [12], including Precision at rank n (P@n), Mean Averaged Precision (MAP), Normalized Discounted Cumulative Gain at rank n (NDCG@n), and bpref [11]. P@n measures the percentage of relevant experts in the top n of the retrieved candidate list, which is estimated as P@n $= \sum_{i=1}^{n} R(c_i)/n$, where $R(c_i) = 1$ if the i-th retrieved candidate is relevant to the given query and $R(c_i) = 0$ otherwise. Suppose there are R_n relevant experts, Average Precision is defined as AP$= \sum_{i=1}^{R_n}(\text{P@}i * R(c_i))/R_n$ and MAP is the averaged AP for all queries. Since the relevance labels are binary, therefore, NDCG is defined as NDCG@n $= \sum_{i=1}^{n} R(c_i)/\log_2(i{+}1)/\sum_{i=1}^{n}\left[1/\log_2(i{+}1)\right]$. Also, we consider bpref, which is a summation based measure of the number of relevant documents ranking before irrelevant ones, bpref$= R_n^{-1}\sum_{r=1}^{R_n}(1-\sum_{i=1}^{r}(1-R(c_i))/R_n)$.

Baselines. We compare LE-expert against the following baselines:

- Balog. The expert finding method based on documents [2].
- NMF. We apply nonnegative matrix factorization [13] to the author-term co-occurrence matrix. The ranking of authors is based on the inner product of the corresponding rows and columns of authors and queries.
- LSI. We apply latent semantic index to identify the similarity of the authors and the queries.
- Corank [124]. Co-ranking cannot be directly applied for expert finding since it is query independent. Therefore, we first retrieve relevant documents and then apply co-ranking for each query.
- Embed [101]. A global embedding algorithm was proposed for the task of expert finding.
- JointHyp [25]. JointHyp is a regularization framework for expert finding in heterogeneous information networks. Specifically, information is propagated through the network based on consistency in the network.
- Exact. The relevance sub-network is extracted based on the exact match.
- RankClass. The sub-relevance sub-network is extracted based on query expansion, and rank the candidates by RankClass with only one class.

Table 6.2: Overall evaluation results.

measure	P@5	P@10	P@20	NDCG@5	NDCG@10	NDCG@20	MAP	bpref
BALOG	0.4941	0.3824	0.2853	0.5068	0.4248	0.3416	0.1608	0.8536
NMF	0.3176	0.2706	0.2118	0.3525	0.3075	0.253	0.1151	0.7303
SVD	0.4353	0.3471	0.2912	0.4553	0.3871	0.3336	0.1548	0.7590
CORANK	0.6941	0.5741	0.4235	0.7181	0.6386	0.5024	0.291	0.8843
EMBED	0.0353	0.0294	0.0265	0.0354	0.0317	0.0289	0.005	0.6331
JOINTHYP	0.6235	0.4176	0.2882	0.6447	0.4913	0.3725	0.1579	**0.9704**
EXACT	0.7059	0.5882	0.4529	0.7548	0.6549	0.5361	0.311	0.8676
RankClass	0.7529	0.6647	0.5176	0.7666	0.7026	0.5867	0.3598	0.8981
LE-expert	**0.8118**	**0.7118**	**0.5559**	**0.8027**	**0.7361**	**0.618**	**0.3826**	0.9451

For fair comparison, we use the same leave-one-out cross-validation dataset and report the best performance of each model. The parameter setting of LE-expert is as follows $\beta = \gamma = 0.5$ in (6.2.1) and (6.2.2). We gradually reduce the size of dimension of the local vector space and set $z = \lceil 300/(5m+1) \rceil$ with m being the hierarchy level. For the concepts $\mathcal{C}_m$, the size of the query expansion ($\mathcal{N}_m$) is set to be $k = 30$. The final expansion for each query ($\widetilde{\mathcal{N}}_m$) is set by cross validation. Recall that $\widetilde{\mathcal{N}}_m$ is query expansion set for relevance sub-network construction. It is worth noting that general queries are more likely to have more expansions and specific ones have less expansions.

6.4.2 Experimental Results

Overall Results Analysis. The experimental results of different methods are summarized in Table 6.2. Compared with BALOG, NMF, SVD, and EMBED which only utilize the textual information and the overall number of citations as the prior of each document, as shown in Table 6.2, we can see that the methods that take advantage of the network information, including CORANK, JOINTHYP, EXACT, RankClass, and LE-expert, achieve significantly better results regarding all the evaluation metrics. This result agrees with our argument that the task of expert finding is different from information retrieval and the network structure plays an important role. Moreover, we notice that the precision of Embed [101] is even worse compared with classical embedding methods, such as NMF and SVD. It can be partially explained by the ranking score for a candidate c and a query q can be loosely interpreted as scaling with $\#(c,q)/(\#c\#q)$ [58], which favors candidates with more pure expertise. More specifically, a candidate with only one paper on q would likely to be ranked topmost.

Now we consider the methods taking advantage of the heterogeneous network structure. Comparing CORANK with EXACT, we see that EXACT performs slightly better than CORANK considering Precisions, NDCG's, and MAP. This is because EXACT additionally considers the venue information for ranking. Moreover, LE-expert significantly outperforms EXACT regarding all the evaluation metrics, which serves as evidence that

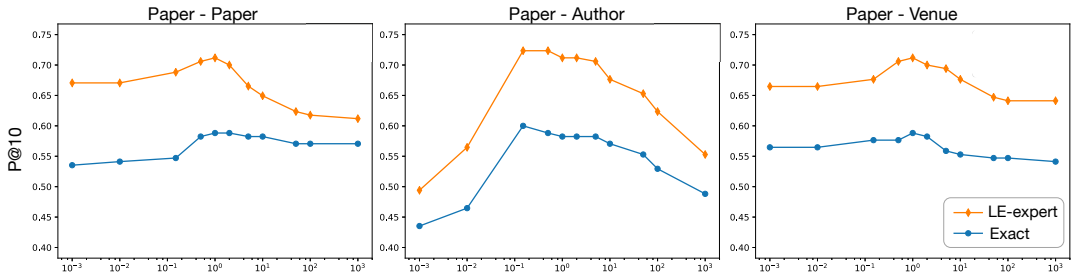


Figure 6.1: The Precision@10 scales with the hyper parameter of the weight of different relation types.

the proposed query expansion method can solve the problem of vocabulary gap. Unlike the global embedding methods (NMF and SVD), LE-expert will not expand specific queries to more general ones thanks to the locally-trained embeddings. LE-expert achieves better precision and NDCG results. JointHyp [26] is also designed for heterogeneous bibliographical information networks, the main idea of which to propagate the relevance of documents for each query to the candidates through the strongly-typed edges in the network. However, such a method will give inaccurate estimation for documents regarding specific queries since the relevance of documents is estimated via global embeddings. Our model is based on the coupled random walks, where the weights for all documents are the same (as $\mathbf{r}_i^0 = 1/|\mathcal{V}_i|$). The prediction accuracy of LE-expert is better than JointHyp; while JointHyp slightly outperforms ours regarding the overall ranking (bref). However, it is worth noting that for the task of expert finding, the top-ranked results are more important. We also compare LE-expert against RankClass, which is similar w.r.t. the ranking algorithm. RankClass is a regularization framework; while LE-expert considers the inter-type and intra-type random walks. We can see that LE-expert performs better than RankClass on precision and NDCG results.

Hyperparameter. As shown in (6.3.1), hyperparameter $\lambda_{\cdot,\cdot}$ (the relative importance of different types of edges) plays an important role for the final ranking of candidates. The sensitivities of the ranking results with varying $\lambda_{\cdot,\cdot}$'s are depicted in Figure 6.1. For simplicity, we set $\lambda_{i,j} = \lambda_{j,i}$ for all i, j. In addition, except for the λ of interest, all the other $\lambda_{\cdot,\cdot} = 1$. The y-axis corresponds to Precision@10. Firstly, we observe that the ranking results are more sensitive to λ_{PA} compared with λ_{PP} and λ_{PV}. This can be explained by the fact that the ranking is based on authors. The second observation is that the precision accuracy follows the pattern of first increasing then decreasing as the weight parameters increase. For one edge type, if the corresponding λ goes to zero, it is equivalent of removing that edge. Such an observation indicates that all the edge types are involved in the final ranking. Our third observation is that when λ_{PP} and λ_{PV} go to zero, the performance remains stable; when λ_{PA} goes to zero, the performance drops significantly. This can be explained by that λ_{PA} balances the relative importance between coauthor relations and writing relations.

Table 6.3: Case study.

boosting		**support vector machine**	
CORANK	LE-expert	CORANK	LE-expert
Robert E. Schapire	**Robert E. Schapire**	Qi Wu	**Bernhard Schölkopf**
Yoav Freund	**Yoav Freund**	Isabelle Guyon	**Vladimir Vapnik**
Ron Kohavi	**Leo Breiman**	**Jason Weston**	**C. J. C. Burges**
Thomas G. Dietterich	**Yoram Singer**	**Vladimir Vapnik**	**Thorsten Joachims**
Yoram Singer	**David P. Helmbold**	Bao-Kiang Lu	**Chih-Jen Lin**
information extraction		**ontology alignment**	
CORANK	LE-expert	CORANK	LE-expert
Ralph Grishman	**Dayne Freitag**	**Jerome Euzenat**	**W. M. Schorlemmer**
Andrew McCallum	**Ralph Grishman**	Patrick Lambrix	**Yannis Kalfoglou**
Ellen Riloff	**Andrew McCallum**	Jason J. Jung	**Anhai Doan**
Oren Etzioni	**Nicholas Kushmerick**	He Tan	**Jerome Euzenat**
Dayne Freitag	**Stephen Soderland**	Marc Ehrig	Alon Y. Halevy

When λ_{PA} goes to zero, the ranking order candidates is dominated by coauthor relations. The absence of authority information from papers leads to fallacious ranking results. Meanwhile, when λ_{PA} goes to infinity, the ranking model is reduced to the document retrieval model (with the relevance of each document to be equal), since the other types of edges do not contribute to the authority scores of candidates.

Case Study. Some concrete case studies of candidate ranking are shown in Table 6.3. For general queries, including "boosting" and "support vector machine", the query expansions are based on the global embeddings. LE-expert has better precision. Particularly, for "support vector machine", "Bernhard Schölkopf", who makes particular contributions to "support vector machine", and "Vladimir Vapnik", who is a co-inventor of the support vector machine, rank topmost. This demonstrates the power of the proposed framework in general queries. For specific queries, we consider "information extraction" (as a child of "natural language processing") and "ontology alignment" (as a child of "ontology"). The high precision results of specific queries indicate that the locally-trained embedding learning method provides accurate and relatively complete expansions for the queries. Moreover, the ranking algorithm contributes to the authority ranking of candidates. Taking "information extraction" as an example, "Dayne Freitag" whose research is focusing on "machine learning for information extraction" ranks higher than more senior researchers "Ralph Grishman" and "Andrew McCallum", given that all of them work on "information extraction".

Chapter 7

Conclusions and Future Work

In this thesis, we studied the problem of identifying the low-dimensional space that high-dimensional data (approximately) lies in. We studied two approaches, one is low-rank estimation models and the other is embedding learning models.

For low-rank estimation models, we established theoretical analysis for both convex and nonconvex regularization.

Regarding convex regularization, we analyzed the statistical performance of robust noisy tensor decomposition with corruption. Our goal is to recover a pair of tensors, based on observing a noisy contaminated version of their sum. It is based on solving a convex optimization with composite regularizations of Schatten-1 norm and ℓ_1 norm defined on tensors. We provided a general nonasymptotic estimator error bounds on the underly low-rank tensor and sparse corruption tensor. Furthermore, the error bound we obtained in this chapter is new, and non-comparable with previous theoretical analysis.

Regarding nonconvex regularization, we proposed a unified framework for low-rank matrix estimation with nonconvex penalty for a generic observation model. Our work serves as the bridge to connect practical applications of nonconvex penalty and theoretical analysis. Our theoretical results indicate that the convergence rate of estimators with nonconvex penalties is faster than the one with the convex penalty by taking advantage of the large singular values. In addition, we showed that the proposed estimator enjoys the oracle property when a mild condition on the magnitude of singular values is imposed. Extensive experiments demonstrate the close agreement between theoretical analysis and numerical behavior of the proposed estimator.

For the second approach of embedding learning models, we studies two different applications, one is object classification in heterogeneous information networks and the other is expert finding.

We proposed to learn object embedding in heterogeneous information networks with events. In detail, we proposed a generic framework called Hebe, which models participant objects in each event as a whole, resulting in more efficient information propagation. Two methods were presented based on the concept of hyperedge: Hebe-Po, modeling the proximity among the participating objects themselves on the same hy-

peredge, and HEBE-PE modeling proximity between the hyperedge and the participating objects. Within the HEBE framework, we presented a parameter-free ranking-based method to efficiently optimize the conditional probabilities via noise sampling. Extensive quantitative experiments have been conducted to corroborate the efficacy of the proposed model in learning the object embeddings, particularly robustness towards noisy observations and data sparseness. We identify some future work for the HEBE framework. Firstly, it is general and could be adapted to many downstream applications, including recommender system and link prediction. Secondly, HEBE prefers term entities from short text due to the operations of subevent sampling. Some additional work needs to be done in order to adapt it to those data having longer text. Thirdly, it could be of interest to learn the relative importance of different event types, based on specific applications. Finally, this work focuses on learning embeddings in an unsupervised manner. Exploring how to incorporate labels and generate predictive embeddings is a another promising direction.

For the application of expert finding, a new framework with two phases is introduced. Firstly, we proposed to perform query expansion based on locally-trained embedding learning recursively with a concept hierarchy as guidance. Secondly, we introduced a ranking algorithm on a relevance sub-network to estimate the expertise of the candidates via coupling both inter-type and intra-type random walks. Numerical experimental results on a large-scale heterogeneous bibliographical information network corroborate the effectiveness of the proposed Le-expert. The proposed framework is general and can be applied to other tasks, such as query-answering in online communities or recruiting for open problem solving. Besides, the locally-trained embedding learning with a concept hierarchy as guidance is of independence interest and may be applied for other tasks, such as product recommendation given a product hierarchy. In addition, since our framework requires a concept hierarchy as input, we leave it for future work when the data does not have any concept hierarchy available.

Bibliography

[1] A. Agarwal, S. Negahban, and M. J. Wainwright. Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics*, 40(2):1171–1197, 04 2012.

[2] K. Balog, L. Azzopardi, and M. De Rijke. Formal models for expert finding in enterprise corpora. In *SIGIR*, pages 43–50. ACM, 2006.

[3] K. Balog, Y. Fang, M. de Rijke, P. Serdyukov, and L. Si. Expertise retrieval. *Foundations and Trends in Information Retrieval*, 6(2–3):127–256, 2012.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2001.

[5] A. R. Benson, D. F. Gleich, and J. Leskovec. Tensor spectral clustering for partitioning higher-order network structures. In *ICDM*, pages 118–126, 2015.

[6] C. Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[7] S. Bhagat, G. Cormode, and S. Muthukrishnan. Node classification in social networks. In *Social network data analytics*, pages 115–148. 2011.

[8] J. Bian, Y. Liu, D. Zhou, E. Agichtein, and H. Zha. Learning to recognize reliable users and content in social media with coupled mutual reinforcement. In *WWW*, 2009.

[9] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, pages 1705–1732, 2009.

[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[11] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *SIGIR*, pages 25–32. ACM, 2004.

[12] S. Büttcher, C. L. Clarke, and G. V. Cormack. *Information retrieval: Implementing and evaluating search engines*. Mit Press, 2016.

[13] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized nonnegative matrix factorization for data representation. *PAMI*, 33(8):1548–1560, 2011.

[14] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[15] T. T. Cai and W. Zhou. Matrix completion via max-norm constrained optimization. *arXiv preprint arXiv:1303.0341*, 2013.

[16] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *J. ACM*, 58(3):11, 2011.

[17] E. J. Candès and Y. Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[18] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Commun. ACM*, 55(6):111–119, 2012.

[19] E. J. Candès and T. Tao. The power of convex relaxation: near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

[20] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

[21] S. Chang, W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, and T. S. Huang. Heterogeneous network embedding via deep architectures. In *KDD*, pages 119–128, 2015.

[22] T. Chen, L.-A. Tang, Y. Sun, Z. Chen, and K. Zhang. Entity embedding-based anomaly detection for heterogeneous categorical events. *Proceeding of 25th International Joint Conference on Artificial Intelligence*, 2016.

[23] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML*, pages 233–240. ACM, 2006.

[24] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JASIST*, 41(6):391, 1990.

[25] H. Deng, J. Han, M. R. Lyu, and I. King. Modeling and exploiting heterogeneous bibliographic networks for expertise ranking. In *ICDL*, pages 71–80. ACM, 2012.

[26] H. Deng, I. King, and M. R. Lyu. Formal models for expert finding on dblp bibliography data. In *ICDM*, pages 163–172. IEEE, 2008.

[27] F. Diaz, B. Mitra, and N. Craswell. Query expansion with locally-trained word embeddings. *arXiv preprint arXiv:1605.07891*, 2016.

[28] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[29] H. Fang and C. Zhai. Probabilistic models for expert finding. In *ECIR*, 2007.

[30] A. Globerson, G. Chechik, F. Pereira, and N. Tishby. Euclidean embedding of co-occurrence data. In *Advances in neural information processing systems*, pages 497–504, 2004.

[31] Y. Goldberg and O. Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[32] P. Gong, J. Ye, and C. Zhang. Robust multi-task feature learning. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 895–903. ACM, 2012.

[33] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *KDD*, 2016.

[34] S. Gunasekar, P. Ravikumar, and J. Ghosh. Exponential family matrix completion under structural constraints. In *Proceedings of the 31st Annual International Conference on Machine Learning*, pages 1917–1925, 2014.

[35] E. T. Hale, W. Yin, and Y. Zhang. Fixed-point continuation for \ell_1-minimization: Methodology and convergence. *SIAM Journal on Optimization*, 19(3):1107–1130, 2008.

[36] M. Hardt. Understanding alternating minimization for matrix completion. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 651–660. IEEE, 2014.

[37] M. Hertzum and A. M. Pejtersen. The information-seeking practices of engineers: searching for documents as well as for people. *Information Processing & Management*, 2000.

[38] D. Hsu, S. M. Kakade, and T. Zhang. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory*, 57(11):7221–7234, 2011.

[39] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. Fast and accurate matrix completion via truncated nuclear norm regularization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(9):2117–2130, 2013.

[40] M. Jaggi and M. Sulovský. A simple algorithm for nuclear norm regularized problems. In *Proceedings of the 27th international conference on machine learning*, pages 471–478, 2010.

[41] P. Jain, R. Meka, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems*, pages 937–945, 2010.

[42] P. Jain and P. Netrapalli. Fast exact matrix completion with finite samples. *arXiv preprint arXiv:1411.1087*, 2014.

[43] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Symposium on Theory of Computing Conference*, pages 665–674, 2013.

[44] R. Jenatton, N. L. Roux, A. Bordes, and G. R. Obozinski. A latent factor model for highly multi-relational data. In *NIPS*, pages 3167–3175, 2012.

[45] M. Ji, J. Han, and M. Danilevsky. Ranking-based classification of heterogeneous information networks. In *KDD*, pages 1298–1306, 2011.

[46] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 457–464, 2009.

[47] M. Jiang, P. Cui, F. Wang, X. Xu, W. Zhu, and S. Yang. Fema: flexible evolutionary multi-faceted analysis for dynamic behavioral pattern discovery. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1186–1195. ACM, 2014.

[48] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[49] M. Karimzadehgan, R. W. White, and M. Richardson. Enhancing expert finding using organizational hierarchies. In *ECIR*, pages 177–188. Springer, 2009.

[50] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[51] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[52] V. Koltchinskii et al. Von neumann entropy penalization and low-rank matrix estimation. *The Annals of Statistics*, 39(6):2936–2973, 2011.

[53] V. Koltchinskii, K. Lounici, A. B. Tsybakov, et al. Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. *The Annals of Statistics*, 39(5):2302–2329, 2011.

[54] Y. Koren. The bellkor solution to the netflix grand prize. 2009.

[55] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[56] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. On the best rank-1 and rank-(r1,r2,. . .,rn) approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.*, 21(4):1324–1342, 2000.

[57] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2001.

[58] O. Levy and Y. Goldberg. Neural word embedding as implicit matrix factorization. In *NIPS*, pages 2177–2185, 2014.

[59] J. Li, A. Ritter, and D. Jurafsky. Learning multi-faceted representations of individuals from heterogeneous evidence using neural networks. *arXiv preprint arXiv:1510.05198*, 2015.

[60] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology*, 58(7):1019–1031, 2007.

[61] D. Liu, T. Zhou, H. Qian, C. Xu, and Z. Zhang. A nearly unbiased matrix completion approach. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 210–225, 2013.

[62] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):208–220, 2013.

[63] J. Liu, X. Ren, J. Shang, T. Cassidy, C. R. Voss, and J. Han. Representing documents via latent keyphrase inference. In *WWW*, 2016.

[64] X. Liu, W. B. Croft, and M. Koll. Finding experts in community-based question-answering services. In *CIKM*, pages 315–316. ACM, 2005.

[65] P.-L. Loh and M. J. Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. In *Advances in Neural Information Processing Systems*, pages 476–484, 2013.

[66] C. Lu, J. Tang, S. Yan, and Z. Lin. Generalized nonconvex nonsmooth low-rank minimization. In *2014 IEEE Conference on CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 4130–4137, 2014.

[67] R. Mazumder, T. Hastie, and R. Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, 11:2287–2322, 2010.

[68] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.

[69] A. Mnih and Y. W. Teh. A fast and simple algorithm for training neural probabilistic language models. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1751–1758, 2012.

[70] C. Mu, B. Huang, J. Wright, and D. Goldfarb. Square deal: Lower bounds and improved relaxations for tensor recovery. *CoRR*, abs/1307.5870, 2013.

[71] S. Negahban and M. J. Wainwright. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, 39(2):1069–1097, 04 2011.

[72] S. Negahban and M. J. Wainwright. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research*, 13:1665–1697, 2012.

[73] S. N. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 11 2012.

[74] F. Nie, H. Wang, X. Cai, H. Huang, and C. H. Q. Ding. Robust matrix completion via joint schatten p-norm and lp-norm minimization. In *IEEE 12th International Conference on Data Mining*, pages 566–574, 2012.

[75] M. G. Noll, C.-m. Au Yeung, N. Gibbins, C. Meinel, and N. Shadbolt. Telling experts from spammers: expertise ranking in folksonomies. In *SIGIR*, pages 612–619. ACM, 2009.

[76] T. Opsahl and P. Panzarasa. Clustering in weighted networks. *Social networks*, 31(2):155–163, 2009.

[77] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.

[78] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *SIGKDD*, pages 701–710. ACM, 2014.

[79] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[80] B. Recht, C. Re, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *NIPS*, pages 693–701, 2011.

[81] S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 81–90. ACM, 2010.

[82] A. Rohde, A. B. Tsybakov, et al. Estimation of high-dimensional low-rank matrices. *The Annals of Statistics*, 39(2):887–930, 2011.

[83] S. Shalev-Shwartz, A. Gonen, and O. Shamir. Large-scale convex minimization with a low-rank constraint. In *Proceedings of the 28th Annual International Conference on Machine Learning*, pages 329–336, 2011.

[84] J. Silva and R. Willett. Hypergraph-based anomaly detection of high-dimensional co-occurrences. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(3):563–569, 2009.

[85] I. Soboroff, A. P. de Vries, and N. Craswell. Overview of the trec 2006 enterprise track. In *Trec*, 2006.

[86] N. Srebro, J. D. M. Rennie, and T. Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems*, pages 1329–1336, 2004.

[87] N. Srebro and A. Shraibman. Rank, trace-norm and max-norm. In *Proceedings of the 18th Annual Conference on Learning Theory*, pages 545–560. Springer-Verlag, 2005.

[88] N. Srebro and A. Shraibman. Rank, trace-norm and max-norm. In *International Conference on Computational Learning Theory*, pages 545–560, 2005.

[89] Y. Sun and J. Han. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 3(2):1–159, 2012.

[90] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *EDBT*, pages 565–576. ACM, 2009.

[91] Y. Sun, Y. Yu, and J. Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *KDD*, pages 797–806, 2009.

[92] J. Tang, M. Qu, and Q. Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *KDD*, pages 1165–1174, 2015.

[93] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1067–1077. ACM, 2015.

[94] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[95] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization*, 6(615-640):15, 2010.

[96] R. Tomioka, K. Hayashi, and H. Kashima. Estimation of low-rank tensors via convex optimization. 2010.

[97] R. Tomioka and T. Suzuki. Convex tensor decomposition via structured schatten norm regularization. In *Advances in Neural Information Processing Systems*, pages 1331–1339, 2013.

[98] R. Tomioka, T. Suzuki, K. Hayashi, and H. Kashima. Statistical performance of convex tensor decomposition. In *Advances in Neural Information Processing Systems*, pages 972–980, 2011.

[99] M. Udell, C. Horn, R. Zadeh, and S. Boyd. Generalized low rank models. *arXiv preprint arXiv:1410.0342*, 2014.

[100] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *JMLR*, 9(2579-2605):85, 2008.

[101] C. Van Gysel, M. de Rijke, and M. Worring. Unsupervised, efficient and semantic expertise retrieval. In *WWW*, 2016.

[102] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV (1)*, pages 447–460, 2002.

[103] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

[104] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959, 2005.

[105] M. E. Wall, A. Rechtsteiner, and L. M. Rocha. Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis*, pages 91–109. Springer, 2003.

[106] G. Wang, Q. Hu, and P. S. Yu. Influence and similarity on heterogeneous networks. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1462–1466. ACM, 2012.

[107] G. A. Wang, J. Jiao, A. S. Abrahams, W. Fan, and Z. Zhang. Expertrank: A topic-aware expert finding algorithm for online knowledge communities. *Decision Support Systems*, 2013.

[108] S. Wang, D. Liu, and Z. Zhang. Nonconvex relaxation approaches to robust matrix recovery. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2013.

[109] Z. Wang, M. Lai, Z. Lu, W. Fan, H. Davulcu, and J. Ye. Rank-one matrix pursuit for matrix completion. In *Proceedings of the 31st Annual International Conference on Machine Learning*, pages 91–99, 2014.

[110] Z. Wang, H. Liu, and T. Zhang. Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *arXiv preprint arXiv:1306.4960*, 2013.

[111] H. Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen. *Mathematische Annalen*, 71(4):441–479, 1912.

[112] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *Advances in neural information processing systems*, pages 2080–2088, 2009.

[113] L. Xiao and T. Zhang. A proximal-gradient homotopy method for the sparse least-squares problem. *SIAM Journal on Optimization*, 23(2):1062–1091, 2013.

[114] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *SIGIR*, pages 4–11. ACM, 1996.

[115] E. Yang and P. D. Ravikumar. Dirty statistical models. In *NIPS*, pages 611–619, 2013.

[116] K.-H. Yang, Y.-L. Lin, and C.-T. Chuang. Using google distance for query expansion in expert finding. In *ICDIM*, pages 104–109. IEEE, 2014.

[117] Q. Yao, J. T. Kwok, and W. Zhong. Fast low-rank matrix learning with nonconvex regularization. *arXiv preprint arXiv:1512.00984*, 2015.

[118] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22(2):179–214, 2004.

[119] C. Zhang, K. Zhang, Q. Yuan, H. Peng, Y. Zheng, T. Hanratty, S. Wang, and J. Han. Regions, periods, activities: Uncovering urban dynamics via cross-modal representation learning. *Proceeding of 26th World Wide Web Conference*, 2017.

[120] C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, pages 894–942, 2010.

[121] C.-H. Zhang, T. Zhang, et al. A general theory of concave regularization for high-dimensional sparse estimation problems. *Statistical Science*, 27(4):576–593, 2012.

[122] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma. Collaborative knowledge base embedding for recommender systems. *Proceeding of 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2016.

[123] J. Zhang, J. Tang, and J. Li. Expert finding in a social network. In *International Conference on Database Systems for Advanced Applications*, pages 1066–1069. Springer, 2007.

[124] D. Zhou, S. A. Orshanskiy, H. Zha, and C. L. Giles. Co-ranking authors and documents in a heterogeneous network. In *ICDM*, 2007.

[125] H. Zou. The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, 101:1418–1429, December 2006.

Appendix A

Proof of Chapter 3

A.1 Proof of Lemma 3.2.1

Proof. Since $\widehat{\mathcal{W}}, \widehat{\mathcal{V}}$ is an optimal solution of (3.2.2), the following inequality holds for any $\mathcal{W}, \mathcal{V} \in \mathbb{R}^{n_1 \times \cdots \times n_K}$:

$$\begin{aligned}
&\frac{1}{2M}\|\mathbf{y} - \mathfrak{X}(\widehat{\mathcal{W}} + \widehat{\mathcal{V}})\|_2^2 + \lambda_M \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} + \mu_M \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \\
&\leq \frac{1}{2M}\|\mathbf{y} - \mathfrak{X}(\mathcal{W}^* + \mathcal{V}^*)\|_2^2 + \lambda_M |||\mathcal{W}^*|||_{S_1} + \mu_M |||\mathcal{V}^*|||_1
\end{aligned}$$

By substituting (4.1.1) into the above equation, and rearranging the items, we have

$$\begin{aligned}
&\sum_{i=1}^{M} \frac{1}{2M}\|y_i^* - \langle \widehat{\mathcal{W}} + \widehat{\mathcal{V}}, \mathcal{X}_i\rangle\|_2^2 \\
&= \sum_{i=1}^{M} \frac{1}{2M}\|y_i^* - \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i\rangle\|_2^2 + \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \right) \\
&- \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i\rangle + \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{W}} + \widehat{\mathcal{V}}, \mathcal{X}_i\rangle \\
&= \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \right) \\
&+ \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{W}} - \mathcal{W}^*, \mathcal{X}_i\rangle + \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{V}} - \mathcal{V}^*, \mathcal{X}_i\rangle \\
&\leq \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \right) \\
&+ \frac{|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\text{mean}}}{M} \left|\!\left|\!\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\!\right|\!\right|_{S_1} + \frac{|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\infty}}{M} \left|\!\left|\!\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\!\right|\!\right|_1 \\
&\leq \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \right) + \frac{\lambda_M}{2} \left|\!\left|\!\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\!\right|\!\right|_{S_1} + \frac{\mu_M}{2} \left|\!\left|\!\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\!\right|\!\right|_1
\end{aligned}$$

Since $\sum_{i=1}^{M} \frac{1}{2M}\|y_i^* - \langle \widehat{\mathcal{W}} + \widehat{\mathcal{V}}, \mathcal{X}_i\rangle\|_2^2 \geq 0$, we have

$$\lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|\!\left|\!\left|\widehat{\mathcal{W}}\right|\!\right|\!\right|_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|\!\left|\!\left|\widehat{\mathcal{V}}\right|\!\right|\!\right|_1 \right) + \frac{\lambda_M}{2} \left|\!\left|\!\left|\widehat{\mathcal{W}} - \mathcal{W}^*\right|\!\right|\!\right|_{S_1} + \frac{\mu_M}{2} \left|\!\left|\!\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\!\right|\!\right|_1 \geq 0$$

which yields

$$\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\Delta}_k''\|_{S_1}+\left|\left|\left|\widehat{\mathcal{V}}_{S^C}\right|\right|\right|_1\leq 3\left(\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\Delta}_k'\|_{S_1}+\left|\left|\left|\widehat{\mathcal{V}}_{S}\right|\right|\right|_1\right)$$

Therefore, there exist $\beta_1 \geq 3$ and $\beta_2 \geq 3$ such that

$$\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\Delta}_k''\|_{S_1}\leq \beta_1\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\Delta}_k'\|_{S_1}$$

$$\left|\left|\left|\mathcal{D}_{S^c}\right|\right|\right|_1\leq \boldsymbol{\beta_2}\left|\left|\left|\mathcal{D}_S\right|\right|\right|_1$$

□

A.2 Proof of Theorem 3.2.2

Proof. Since $\widehat{\mathcal{W}}, \widehat{\mathcal{V}}$ is an optimal solution of (3.2.2), the following inequality holds for any $\mathcal{W}, \mathcal{V} \in \mathbb{R}^{n_1\times\ldots\times n_K}$:

$$\begin{aligned}&\frac{1}{2M}\|\mathbf{y}-\mathfrak{X}(\widehat{\mathcal{W}}+\widehat{\mathcal{V}})\|_2^2+\lambda_M\left|\left|\left|\widehat{\mathcal{W}}\right|\right|\right|_{S_1}+\mu_M\left|\left|\left|\widehat{\mathcal{V}}\right|\right|\right|_1\\&\leq\frac{1}{2M}\|\mathbf{y}-\mathfrak{X}(\mathcal{W}^*+\mathcal{V}^*)\|_2^2+\lambda_M\left|\left|\left|\mathcal{W}^*\right|\right|\right|_{S_1}+\mu_M\left|\left|\left|\mathcal{V}^*\right|\right|\right|_1\end{aligned}$$

By substituting (4.1.1) into the above equation, and rearranging the items, we have

$$
\begin{aligned}
&\sum_{i=1}^{M} \frac{1}{2M} \|y_i^* - \langle \widehat{\mathcal{W}} + \widehat{\mathcal{V}}, \mathcal{X}_i \rangle\|_2^2 \\
&= \sum_{i=1}^{M} \frac{1}{2M} \|y_i^* - \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i \rangle\|_2^2 + \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|||\widehat{\mathcal{W}}\right|||_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|||\widehat{\mathcal{V}}\right|||_1 \right) \\
&- \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i \rangle + \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{W}} + \widehat{\mathcal{V}}, \mathcal{X}_i \rangle \\
&= \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|||\widehat{\mathcal{W}}\right|||_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|||\widehat{\mathcal{V}}\right|||_1 \right) \\
&+ \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{W}} - \mathcal{W}^*, \mathcal{X}_i \rangle + \sum_{i=1}^{M} \frac{\epsilon_i}{M} \langle \widehat{\mathcal{V}} - \mathcal{V}^*, \mathcal{X}_i \rangle \\
&\leq \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|||\widehat{\mathcal{W}}\right|||_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|||\widehat{\mathcal{V}}\right|||_1 \right) \\
&+ \frac{|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\text{mean}}}{M} \left|||\widehat{\mathcal{W}} - \mathcal{W}^*\right|||_{S_1} + \frac{|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\infty}}{M} \left|||\widehat{\mathcal{V}} - \mathcal{V}^*\right|||_1 \\
&\leq \lambda_M \left(|||\mathcal{W}^*|||_{S_1} - \left|||\widehat{\mathcal{W}}\right|||_{S_1} \right) + \mu_M \left(|||\mathcal{V}^*|||_1 - \left|||\widehat{\mathcal{V}}\right|||_1 \right) + \frac{\lambda_M}{2} \left|||\widehat{\mathcal{W}} - \mathcal{W}^*\right|||_{S_1} + \frac{\mu_M}{2} \left|||\widehat{\mathcal{V}} - \mathcal{V}^*\right|||_1 \\
&\leq \frac{3\lambda_M}{2} \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{\Delta}_k'\|_{S_1} + \frac{3\mu_M}{2} |||\mathcal{D}_S|||_1
\end{aligned}
$$

where the second equality follows from the fact that $y_i^* = \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i \rangle$, the first inequality is obtained by Hölder's inequality and Hölder-like inequality (3.1.1), the second inequality follows from the assumption, and the last equality is obtained from Lemma **??**.

As $y_i^* = \langle \mathcal{W}^* + \mathcal{V}^*, \mathcal{X}_i \rangle$, we have

$$
\frac{1}{2M} \|\mathfrak{X}(\Delta + \mathcal{D})\|_2^2 \leq \frac{3\lambda_M}{2} \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{\Delta}_k'\|_{S_1} + \frac{3\mu_M}{2} |||\mathcal{D}_S|||_1 . \tag{A.2.1}
$$

This completes the proof. □

A.3 Proof of Theorem 3.2.4

Proof. Based on inequality (3.1.2) and Assumption 3.2.3, we have

$$\begin{aligned}\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\Delta}'_k\|_{S_1} &\le \frac{1}{K}\sum_{k=1}^{K}\sqrt{2r_k}\|\boldsymbol{\Delta}'_k\|_F \\ &\le \frac{1}{K}\sum_{k=1}^{K}\sqrt{2r_k}\|\boldsymbol{\Delta}_{(k)}\|_F \\ &= \frac{1}{K}\sum_{k=1}^{K}\sqrt{2r_k}\,|||\Delta|||_F \\ &\le \frac{1}{K}\sum_{k=1}^{K}\frac{\sqrt{2r_k}}{\kappa_1\sqrt{M}}\|\mathfrak{X}(\Delta+\mathcal{D})\|_2 \end{aligned} \tag{A.3.1}$$

and

$$\left|\left|\left|(\widehat{\mathcal{V}}-\mathcal{V}^*)_S\right|\right|\right|_1 \le \sqrt{s}\,|||\mathcal{D}_S|||_F \le \sqrt{s}\,|||\mathcal{D}|||_F \le \frac{\sqrt{s}}{\kappa_2\sqrt{M}}\|\mathfrak{X}(\Delta+\mathcal{D})\|_2 \tag{A.3.2}$$

Substituting these inequalities into Theorem 3.2.2,

$$\begin{aligned}\frac{1}{2M}\|\mathfrak{X}(\Delta+\mathcal{D})\|_2^2 &\le \frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1\sqrt{M}}\|\mathfrak{X}(\Delta+\mathcal{D})\|_2 + \frac{3\mu_M\sqrt{s}}{2\kappa_2\sqrt{M}}\|\mathfrak{X}(\Delta+\mathcal{D})\|_2 \\ &= \left(\frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1\sqrt{M}} + \frac{3\mu_M\sqrt{s}}{2\kappa_2\sqrt{M}}\right)\|\mathfrak{X}(\Delta+\mathcal{D})\|_2.\end{aligned}$$

So that $|||\mathfrak{X}(\Delta+\mathcal{D})|||_2$ is bounded as follows:

$$\|\mathfrak{X}(\Delta+\mathcal{D})\|_2 \le 2\sqrt{M}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right).$$

Thus, by Assumption 3.2.3, we obtain

$$\begin{aligned}\left|\left|\left|\widehat{\mathcal{W}}-\mathcal{W}^*\right|\right|\right|_F &\le \frac{2}{\kappa_1}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right) \\ \left|\left|\left|\widehat{\mathcal{V}}-\mathcal{V}^*\right|\right|\right|_F &\le \frac{2}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right)\end{aligned}$$

This completes the proof. □

A.4 Proof of Corollary 3.2.5

Although the Frobenius norm-based estimation error bounds of $\widehat{\mathcal{W}}$ and $\widehat{\mathcal{V}}$ are mostly useful, it is also interesting to extend the results to Schatten-1 norm and ℓ_1-norm-based estimation error bounds, as stated in the following corollary.

Corollary A.4.1. *Under the same conditions of Theorem 3.2.4, we have*

$$\left|\left|\left|\widehat{\mathcal{W}}-\mathcal{W}^*\right|\right|\right|_{S_1} \leq \frac{2(1+\beta_1)\sum_{k=1}^K\sqrt{2r_k}}{\kappa_1 K}\left(\frac{1}{K}\sum_{k=1}^K\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1}+\frac{3\mu_M\sqrt{s}}{2\kappa_2}\right),$$
$$\left|\left|\left|\widehat{\mathcal{V}}-\mathcal{V}^*\right|\right|\right|_1 \leq \frac{2(1+\beta_2)\sqrt{s}}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^K\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1}+\frac{3\mu_M\sqrt{s}}{2\kappa_2}\right).$$

Proof. According to Assumption 3.2.3 that $\sum_{k=1}^K\left|\left|\left|\mathbf{\Delta}_k''\right|\right|\right|_{S_1}\leq\beta_1\sum_{k=1}^K\left|\left|\left|\mathbf{\Delta}_k'\right|\right|\right|_{S_1}$, we have

$$\begin{aligned}
\left|\left|\left|\widehat{\mathcal{W}}-\mathcal{W}^*\right|\right|\right|_{S_1} &= \frac{1}{K}\left(\sum_{k=1}^K\|\mathbf{\Delta}_k'\|_{S_1}+\sum_{k=1}^K\|\mathbf{\Delta}_k''\|_{S_1}\right)\\
&\leq \frac{(1+\beta_1)}{K}\sum_{k=1}^K\|\mathbf{\Delta}_k'\|_{S_1}\\
&\leq \frac{(1+\beta_1)}{K}\sum_{k=1}^K\sqrt{2r_k}\left|\left|\left|\Delta\right|\right|\right|_F\\
&\leq \frac{2(1+\beta_1)\sum_{k=1}^K\sqrt{2r_k}}{\kappa_1 K}\left(\frac{1}{K}\sum_{k=1}^K\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1}+\frac{3\mu_M\sqrt{s}}{2\kappa_2}\right)
\end{aligned}$$

For the second part,

$$\begin{aligned}
\left|\left|\left|\mathcal{D}\right|\right|\right|_1 &= \left|\left|\left|\mathcal{D}_S\right|\right|\right|_1+\left|\left|\left|\mathcal{D}_{S^c}\right|\right|\right|_1\\
&\leq (1+\beta_2)\left|\left|\left|\mathcal{D}_S\right|\right|\right|_1\\
&\leq (1+\beta_2)\sqrt{s}\left|\left|\left|\mathcal{D}\right|\right|\right|_F\\
&\leq \frac{2(1+\beta_2)\sqrt{s}}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^K\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1}+\frac{3\mu_M\sqrt{s}}{2\kappa_2}\right)
\end{aligned}$$

□

Proof. (Proof of Corollary 3.2.5)

Proof of part (a): we need to show the following two parts:

a.(i) $r\leq\widehat{r}_k\Rightarrow r\leq r_k$;

a.(ii) $r \leq r_k \Rightarrow r \leq \widehat{r}_k$;

For a.(i), assume $r \leq \widehat{r}_k$ but $r > r_k$, we have

$$\begin{aligned}\sigma_r(\widehat{\mathbf{W}}_{(k)} - \mathbf{W}^*_{(k)}) &\geq \sigma_r(\widehat{\mathbf{W}}_{(k)}) \\ &\geq \sigma_{r_k}(\widehat{\mathbf{W}}_{(k)}) \\ &> \frac{2(1+\beta_1)\sum_{k=1}^K \sqrt{2r_k}}{\kappa_1 MK}\left(\frac{1}{K}\sum_{k=1}^K \frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right),\end{aligned}$$

On the other hand, we have

$$\sigma_r(\widehat{\mathbf{W}}_{(k)} - \mathbf{W}^*_{(k)}) \leq \|\boldsymbol{\Delta}_{(k)}\|_{S_1} \leq \frac{2(1+\beta_1)\sum_{k=1}^K \sqrt{2r_k}}{\kappa_1 MK}\left(\frac{1}{K}\sum_{k=1}^K \frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right), \tag{A.4.1}$$

The above two inequalities contradict with each other, thus a.(i) holds.

For a.(ii), assume $r \leq r_k$ but $r > \widehat{r}_k$, we have

$$\begin{aligned}\sigma_r(\widehat{\mathbf{W}}_{(k)} - \mathbf{W}^*_{(k)}) &\geq \sigma_r(\mathbf{W}^*_{(k)}) - \sigma_r(\widehat{\mathbf{W}}_{(k)}) \\ &\geq \sigma_{r_k}(\mathbf{W}_{(k)^*}) - \sigma_{r_k}(\widehat{\mathbf{W}}_{(k)}) \\ &> \frac{2(1+\beta_1)\sum_{k=1}^K \sqrt{2r_k}}{\kappa_1 MK}\left(\frac{1}{K}\sum_{k=1}^K \frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right),\end{aligned}$$

which contradicts with (A.4.1), thus we verify a.(ii). Therefore, we have $\widehat{r}_k = r_k$.

Proof of part (b): we need to show the following two parts:

b.(i) $\forall (i_1, \ldots, i_K) \in \widehat{S} \Rightarrow (i_1, \ldots, i_K) \in S$;

b.(ii) $\forall (i_1, \ldots, i_K) \in S \Rightarrow (i_1, \ldots, i_K) \in \widehat{S}$;

For b.(i), assume there exists a $(j_1, \ldots, j_K) \in \widehat{S}$, $(j_1, \ldots, j_K) \notin S$. Based on the definition of S and $\widehat{S}$, we have

$$|(\widehat{\mathcal{V}} - \mathcal{V}^*)_{j_1,\ldots,j_K}| = |\widehat{\mathcal{V}}_{j_1,\ldots,j_K}| > \frac{2(1+\beta_2)\sqrt{s}}{\kappa_2}\left(\frac{1}{K}\sum_{k=1}^K \frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right).$$

On the other hand, we have

$$|(\widehat{\mathcal{V}} - \mathcal{V}^*)_{j_1,\ldots,j_K}| \leq \left|\left|\left|\widehat{\mathcal{V}} - \mathcal{V}^*\right|\right|\right|_1 \leq \frac{2(1+\beta_2)\sqrt{s}}{\kappa_2 M}\left(\frac{1}{K}\sum_{k=1}^K \frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right). \tag{A.4.2}$$

The above two inequalities contradict with each other, thus b.(i) holds.

Similarly, we can verify b.(ii), assume there exists a $(j'_1, \ldots, j'_K) \in S$, $(j'_1, \ldots, j'_K) \notin \widehat{S}$, we have

$$|(\widehat{\mathcal{V}} - \mathcal{V}^*)_{j'_1,\ldots,j'_K}| \geq |\mathcal{V}^*_{j'_1,\ldots,j'_K}| - |\widehat{\mathcal{V}}_{j'_1,\ldots,j'_K}| > \frac{2(1+\beta_2)\sqrt{s}}{\kappa_2 M}\left(\frac{1}{K}\sum_{k=1}^{K}\frac{3\lambda_M\sqrt{2r_k}}{2\kappa_1} + \frac{3\mu_M\sqrt{s}}{2\kappa_2}\right),$$

which contradicts with (A.4.2), thus we verify b.(ii). Therefore, we have $\widehat{S} = S$. □

A.5 Proof of Lemma 3.2.6

Proof. For $k = 1, \ldots, K$, $\mathfrak{X}^*(\boldsymbol{\epsilon})_{(k)}$ is a $n_k \times \bar{N}_{\backslash k}$ matrix, whose entries follow distribution of $N(0, \sigma)$, $\left|\left|\left|\mathfrak{X}^*(\boldsymbol{\epsilon})_{(k)}\right|\right|\right|_{S_\infty}$ are Lipschitz function with Lipschitz constant $L = 1$, so that In addition, we have $|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_{\text{mean}}$ and $|||\mathfrak{X}^*(\boldsymbol{\epsilon})|||_\infty$ are Lipschitz function with Lipschitz constant $L = 1$. According to [103], Proposition 5.34 and Corollary 5.35, we obtain the claim. □

Appendix B

Proof of Chapter 4

B.1 Introduction

In this supplement, we first provide additional experimental results on the proposed estimator with MCP regularization, followed by the details of technical proof for the main results, including proofs of theorems and auxiliary lemmas.

B.2 Additional Experimental Results

Regarding matrix completion and matrix sensing, we present additional experimental results of the proposed estimator with MCP penalty. Due to the similar properties and parameter settings of these two nonconvex penalties, the MCP penalty and SCAD penalty, the numerical behaviour of the proposed estimator with MCP penalty resembles the one with SCAD penalty, as shown in Figure B.1.

In detail, Figure B.1(a)- B.1(c) are the results for matrix completion. Accordingly, the size of matrix and the rank are $m \times m$. The results of matrix completion, with rank $r = \lfloor \log^2 m \rfloor$, in Figure B.1(a)-B.1(c) with the rescaled sample size $N = n/(rm \log m)$; while matrix sensing, for rank $r = 10$, is studied in Figure B.1(d)-B.1(f) with rescaled sample size $N = n/(rm)$. With the same settings as experiments shown in Figure 4.1, we have that the estimator with MCP penalty, a particular case of the proposed estimator with nonconvex penalty, behaviors in accordance with our theoretical analysis and outperforms the estimator with nuclear norm. For the other example, *i.e.*, matrix sensing, the results in Figure B.1(d)- B.1(f) manifest the superiority of the estimator with MCP penalty. Particularly, for both examples, we have with with high probability, the rank of the underlying matrix is recovered with high probability.

B.3 Background

For matrix $\mathbf{\Theta}^* \in \mathbb{R}^{m_1 \times m_2}$, which is exactly low-rank and has rank r, we have the singular value decomposition (SVD) form of $\mathbf{\Theta}^* = \mathbf{U}^* \mathbf{\Gamma}^* \mathbf{V}^{*\top}$, where $\mathbf{U}^* \in \mathbb{R}^{m_1 \times r}$, $\mathbf{V}^* \in \mathbb{R}^{m_2 \times r}$ are matrices consist of left and right

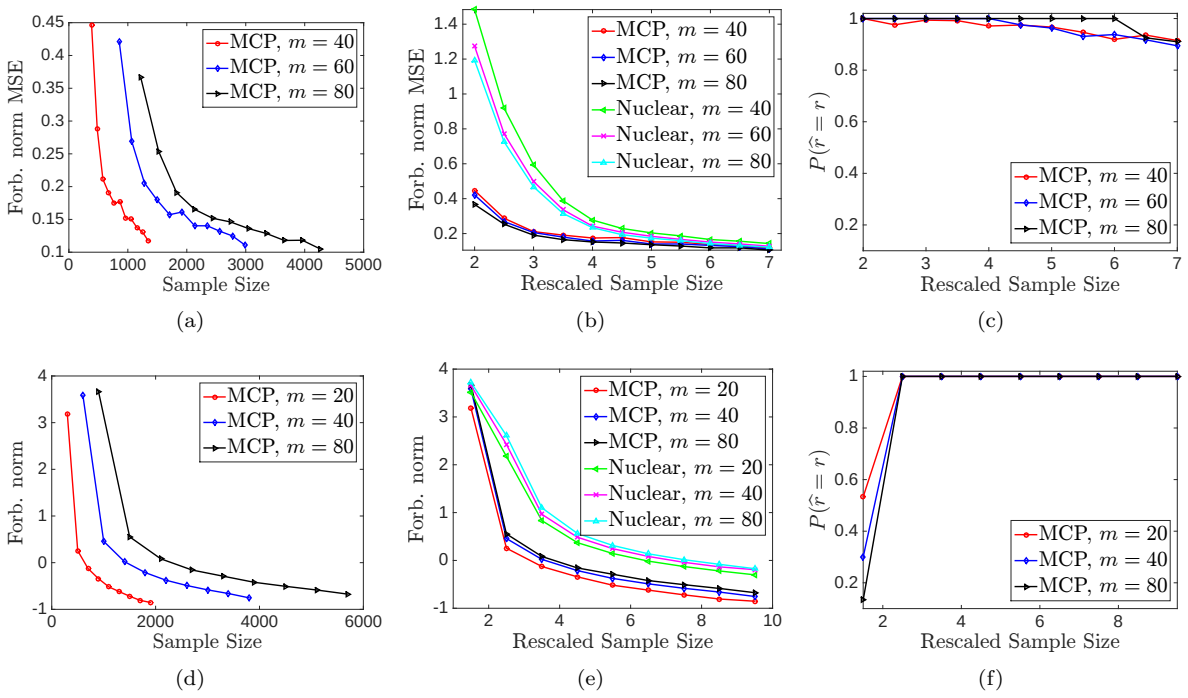


Figure B.1: Simulation Results for Matrix Completion and Matrix Sensing with MCP penalty.

singular vectors, and $\mathbf{\Gamma}^* = \mathrm{diag}(\gamma_1^*, \ldots, \gamma_r^*) \in \mathbb{R}^{r \times r}$. Based on $\mathbf{U}^*, \mathbf{V}^*$, we define the following two subspaces of $\mathbb{R}^{m_1 \times m_2}$:

$$\mathcal{F}(\mathbf{U}^*, \mathbf{V}^*) := \{\mathbf{\Delta} | \mathrm{row}(\mathbf{\Delta}) \subseteq \mathbf{V}^* \text{ and } \mathrm{col}(\mathbf{\Delta}) \subseteq \mathbf{U}^*\},$$

and

$$\mathcal{F}^{\perp}(\mathbf{U}^*, \mathbf{V}^*) := \{\mathbf{\Delta} | \mathrm{row}(\mathbf{\Delta}) \perp \mathbf{V}^* \text{ and } \mathrm{col}(\mathbf{\Delta}) \perp \mathbf{U}^*\},$$

where $\mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2}$ is an arbitrary matrix, and $\mathrm{row}(\mathbf{\Delta}) \subseteq \mathbb{R}^{m_2}$, $\mathrm{col}(\mathbf{\Delta}) \subseteq \mathbb{R}^{m_1}$ are the row space and column space of the matrix $\mathbf{\Delta}$. respectively. We will use the shorthand notation of $\mathcal{F}, \mathcal{F}^{\perp}$, when $(\mathbf{U}^*, \mathbf{V}^*)$ are clear from the context. Define $\mathbf{\Pi}_{\mathcal{F}}, \mathbf{\Pi}_{\mathcal{F}^{\perp}}$ as the projection operator onto the subspaces $\mathcal{F}$ and $\mathcal{F}^{\perp}$:

$$\begin{aligned} \mathbf{\Pi}_{\mathcal{F}}(\mathbf{A}) &= \mathbf{U}^* \mathbf{U}^{*\top} \mathbf{A} \mathbf{V}^* \mathbf{V}^{*\top}, \\ \mathbf{\Pi}_{\mathcal{F}^{\perp}}(\mathbf{A}) &= \big(\mathbf{I}_{m_1} - \mathbf{U}^* \mathbf{U}^{*\top}\big) \mathbf{A} \big(\mathbf{I}_{m_2} - \mathbf{V}^* \mathbf{V}^{*\top}\big). \end{aligned} \tag{B.3.1}$$

Thus, for all $\mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2}$, we have its orthogonal complement $\mathbf{\Delta}''$ with respect to the true low-rank matrix $\mathbf{\Theta}^*$ as follows:

$$\begin{aligned} \mathbf{\Delta}'' =& \big(\mathbf{I}_{m_1} - \mathbf{U}^* \mathbf{U}^{*\top}\big) \mathbf{\Delta} \big(\mathbf{I}_{m_2} - \mathbf{V}^* \mathbf{V}^{*\top}\big), \\ \mathbf{\Delta}' =& \mathbf{\Delta} - \mathbf{\Delta}'', \end{aligned} \tag{B.3.2}$$

where $\mathbf{\Delta}'$ is the component which has overlapped row and column space with $\mathbf{\Theta}^*$. [73] gives detailed discussion about the concept of decomposibility and a large class of decomposable norms, among which the decomposability of the nuclear norm and Frobenius norm is relevant to our problem. For low-rank estimation, we have the equality that $\|\mathbf{\Theta}^* + \mathbf{\Delta}''\|_* = \|\mathbf{\Theta}^*\|_* + \|\mathbf{\Delta}''\|_*$ with $\mathbf{\Delta}''$ defined above.

B.4 Proof of the Main Results

In this section, we provide detailed proof for the main results.

B.4.1 Proof of Theorem 4.2.4

We first define $\widetilde{\mathcal{L}}_{n,\lambda}(\cdot)$ as follows,

$$\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) = \mathcal{L}_n(\mathbf{\Theta}) + \mathcal{Q}_\lambda(\mathbf{\Theta}).$$

Based on the the restrict strongly convexity of $\mathcal{L}_n$, and the curvature parameter of the non-convex penalty, if $\kappa(\mathfrak{X}) > \zeta_-$, we have the restrict strongly convexity of $\widetilde{\mathcal{L}}_{n,\lambda}(\cdot)$, as stated in the following lemma.

Lemma B.4.1. *Under Assumption 4.2.1, if it is assumed that $\mathbf{\Theta}_1 - \mathbf{\Theta}_2 \in \mathcal{C}$, we have*

$$\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}_2) \geq \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}_1) + \langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}_1), \mathbf{\Theta}_2 - \mathbf{\Theta}_1 \rangle + \frac{\kappa(\mathfrak{X}) - \zeta_-}{2} \|\mathbf{\Theta}_2 - \mathbf{\Theta}_1\|_F^2.$$

Proof. Proof is provided in Section B.6.1. □

In the following, we prove that $\widehat{\mathbf{\Delta}} = \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*$ lies in the cone $\mathcal{C}$, where

$$\mathcal{C} = \big\{ \mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2} \big| \|\Pi_{\mathcal{F}^\perp}(\mathbf{\Delta})\|_* \leq 5 \|\Pi_{\mathcal{F}}(\mathbf{\Delta})\|_* \big\}.$$

Lemma B.4.2. *Under Assumption 4.2.1, the condition $\kappa(\mathfrak{X}) > \zeta_-$, and the regularization parameter $\lambda \geq$*

$2\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2/n$, *we have*

$$\|\Pi_{\mathcal{F}}(\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*)\|_* \leq 5\|\Pi_{\mathcal{F}^\perp}(\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*)\|_*.$$

Proof. Proof is provided in Section B.6.2. □

Now we are ready to prove Theorem 4.2.4.

Proof of Theorem 4.2.4. According to Lemma B.4.1, we have

$$\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) \geq \widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^*) + \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^*), \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^* \rangle + \frac{\kappa(\mathfrak{X}) - \zeta_-}{2}\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*\|_F^2, \tag{B.4.1}$$

$$\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^*) \geq \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) + \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}), \boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}} \rangle + \frac{\kappa(\mathfrak{X}) - \zeta_-}{2}\|\boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}}\|_F^2. \tag{B.4.2}$$

Meanwhile, since $\|\cdot\|_*$ is convex, we have

$$\lambda\|\widehat{\boldsymbol{\Theta}}\|_* \geq \lambda\|\boldsymbol{\Theta}^*\|_* + \lambda\langle \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*, \mathbf{W}^* \rangle, \tag{B.4.3}$$

$$\lambda\|\boldsymbol{\Theta}^*\|_* \geq \lambda\|\widehat{\boldsymbol{\Theta}}\|_* + \lambda\langle \boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}}, \mathbf{W}^* \rangle, \tag{B.4.4}$$

where $\mathbf{W}^* \in \|\boldsymbol{\Theta}^*\|_*$.

Adding (B.4.1) to (B.4.4), we have

$$0 \geq \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^*) + \lambda\mathbf{W}^*, \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^* \rangle + \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) + \lambda\widehat{\mathbf{W}}, \boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}} \rangle + (\kappa(\mathfrak{X}) - \zeta_-)\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^*\|_F^2.$$

Since $\widehat{\boldsymbol{\Theta}}$ is the solution to the SDP (4.1.2), $\widehat{\boldsymbol{\Theta}}$ satisfies the optimality condition (variational inequality), for any $\boldsymbol{\Theta}' \in \mathbb{R}^{m_1 \times m_2}$, it holds that

$$\max_{\boldsymbol{\Theta}'} \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) + \lambda\widehat{\mathbf{W}}, \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}' \rangle \leq 0,$$

which implies

$$\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) + \lambda\widehat{\mathbf{W}}, \boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}} \rangle \geq 0.$$

Hence,

$$
\begin{aligned}
&(\kappa(\mathfrak{X})-\zeta_{-})\|\widehat{\boldsymbol{\Theta}}-\boldsymbol{\Theta}^{*}\|_{F}^{2} \leq \big\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^{*})+\lambda\mathbf{W}^{*}, \boldsymbol{\Theta}^{*}-\widehat{\boldsymbol{\Theta}}\big\rangle \\
&\leq \Big\langle \boldsymbol{\Pi}_{\mathcal{F}^{\perp}}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^{*})+\lambda\mathbf{W}^{*}\big), \boldsymbol{\Theta}^{*}-\widehat{\boldsymbol{\Theta}}\Big\rangle + \Big\langle \boldsymbol{\Pi}_{\mathcal{F}}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\boldsymbol{\Theta}^{*})+\lambda\mathbf{W}^{*}\big), \boldsymbol{\Theta}^{*}-\widehat{\boldsymbol{\Theta}}\Big\rangle. \qquad \text{(B.4.5)}
\end{aligned}
$$

Recall that $\boldsymbol{\gamma}^{*}=\boldsymbol{\gamma}(\boldsymbol{\Theta}^{*})$ is the vector of (ordered) singular values of $\boldsymbol{\Theta}^{*}$. In the following, we decompose (B.4.5) into three parts with regard to the magnitudes of the singular values of $\boldsymbol{\Theta}^{*}$.

(1) $i\in S^{c}$ that $(\boldsymbol{\gamma}^{*})_{i}=0$;

(2) $i\in S_{1}$ that $(\boldsymbol{\gamma}^{*})_{i}\geq\nu$;

(3) $i\in S_{2}$ that $\nu>(\boldsymbol{\gamma}^{*})_{i}>0$.

Note that $S_{1}\cup S_{2}=S$.

(1) For $i\in S^{c}$, it correspond to the projector $\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}(\cdot)$ since $\boldsymbol{\gamma}(\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}(\boldsymbol{\Theta}^{*}))=(\boldsymbol{\gamma}^{*})_{S^{c}}=\mathbf{0}$.

Based on the regularity condition (iii) in Assumption 4.2.3 that $q_{\lambda}'(0)=0$, we have that $\nabla\mathcal{Q}_{\lambda}(\boldsymbol{\Theta}^{*})=\mathbf{U}^{*}q_{\lambda}'(\boldsymbol{\Gamma}^{*})\mathbf{V}^{*\top}$ where $\boldsymbol{\Gamma}^{*}\in\mathbb{R}^{r\times r}$ is the diagonal matrix with $\mathrm{diag}(\boldsymbol{\Gamma}^{*})=\boldsymbol{\gamma}^{*}$, we have

$$
\begin{aligned}
\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}(\nabla\mathcal{Q}_{\lambda}(\boldsymbol{\Theta}^{*})) &= \big(\mathbf{I}_{m_1}-\mathbf{U}^{*}\mathbf{U}^{*\top}\big)\mathbf{U}^{*}q_{\lambda}'(\boldsymbol{\Gamma}^{*})\mathbf{V}^{*\top}\big(\mathbf{I}_{m_2}-\mathbf{V}^{*}\mathbf{V}^{*\top}\big) \\
&= (\mathbf{U}^{*}-\mathbf{U}^{*})q_{\lambda}'(\boldsymbol{\Gamma}^{*})\big(\mathbf{V}^{*\top}-\mathbf{V}^{*\top}\big) \\
&= \mathbf{0}.
\end{aligned}
$$

Therefore,

$$
\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}(\nabla\mathcal{Q}_{\lambda}(\boldsymbol{\Theta}^{*}))=\mathbf{0}.
$$

Meanwhile, we have

$$
\big\|\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}\big(\nabla\mathcal{L}_{n}(\boldsymbol{\Theta}^{*})\big)\big\|_{2}\leq\big\|\nabla\mathcal{L}_{n}(\boldsymbol{\Theta}^{*})\big\|_{2}=\frac{\|\mathfrak{X}^{*}(\boldsymbol{\epsilon})\|_{2}}{n}\leq\lambda.
$$

For $\mathbf{Z}^{*}=-\lambda^{-1}\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}\big(\nabla\mathcal{L}_{n}(\boldsymbol{\Theta}^{*})\big)$, we have $\mathbf{W}^{*}=\mathbf{U}^{*}\mathbf{V}^{*\top}+\mathbf{Z}^{*}\in\partial\|\boldsymbol{\Theta}^{*}\|_{*}$ because $\|\mathbf{Z}^{*}\|_{2}\leq 1$ and $\mathbf{Z}^{*}\in\mathcal{F}^{\perp}$, which satisfies the condition of $\mathbf{W}^{*}$ to be subgradient of $\|\boldsymbol{\Theta}^{*}\|_{*}$. With this particular choice of $\mathbf{W}^{*}$, we have

$$
\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}\big(\nabla\mathcal{L}_{n}(\boldsymbol{\Theta}^{*})+\lambda\mathbf{W}^{*}\big)=\boldsymbol{\Pi}_{\mathcal{F}^{\perp}}\big(\nabla\mathcal{L}_{n}(\boldsymbol{\Theta}^{*})\big)+\lambda\mathbf{Z}^{*}=\mathbf{0},
$$

which implies that

$$\left\langle \mathbf{\Pi}_{\mathcal{F}^\perp}\left(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*)+\lambda\mathbf{W}^*\right), \mathbf{\Theta}^*-\widehat{\mathbf{\Theta}}\right\rangle = \left\langle \mathbf{0}, \mathbf{\Theta}^*-\widehat{\mathbf{\Theta}}\right\rangle = 0. \tag{B.4.6}$$

(2) Consider $i \in S_1$ that $(\boldsymbol{\gamma}^*)_i \geq \nu$. Let $|S_1| = r_1$. Define a subspace of $\mathcal{F}$ associated with S_1 as follows

$$\mathcal{F}_{S_1}(\mathbf{U}^*, \mathbf{V}^*) := \{\mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2} | \text{row}(\mathbf{\Delta}) \subset \mathbf{V}^*_{S_1} \text{ and } \text{col}(\mathbf{\Delta}) \subset \mathbf{U}^*_{S_1}\},$$

where $\mathbf{U}^*_{S_1}$ and $\mathbf{V}^*_{S_1}$ is the matrix with the i^{th} row of $\mathbf{U}^*$ and $\mathbf{V}^*$ where $i \in S_1$.

Recall that $\mathcal{P}_\lambda(\mathbf{\Theta}^*) = \mathcal{Q}_\lambda(\mathbf{\Theta}^*) + \lambda\|\mathbf{\Theta}^*\|_*$. We have

$$\nabla\mathcal{P}_\lambda(\mathbf{\Theta}^*) = \nabla\mathcal{Q}_\lambda(\mathbf{\Theta}^*) + \lambda(\mathbf{U}^*\mathbf{V}^{*\top} + \mathbf{Z}^*).$$

Projecting $\nabla\mathcal{P}_\lambda(\mathbf{\Theta}^*)$ into the subspace $\mathcal{F}_{S_1}$, we have

$$\begin{aligned}
\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{P}_\lambda(\mathbf{\Theta}^*)\right) &= \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{Q}_\lambda(\mathbf{\Theta}^*) + \lambda\mathbf{U}^*\mathbf{V}^{*\top} + \lambda\mathbf{Z}^*\right) \\
&= \mathbf{U}^*_{S_1} q'_\lambda(\mathbf{\Gamma}^*_{S_1})(\mathbf{V}^*_{S_1})^\top + \lambda\mathbf{U}^*_{S_1}(\mathbf{V}^*_{S_1})^\top \\
&= \mathbf{U}^*_{S_1}\left(q'_\lambda(\mathbf{\Gamma}^*_{S_1}) + \lambda\mathbf{I}_{S_1}\right)(\mathbf{V}^*_{S_1})^\top,
\end{aligned}$$

where $\mathbf{\Gamma}^*_{S_1} \in \mathbb{R}^{r_1 \times r_1}$ and $\left(q'_\lambda(\mathbf{\Gamma}^*_{S_1}) + \lambda\mathbf{I}_{S_1}\right)$ is a diagonal matrix that $\left(q'_\lambda(\mathbf{\Gamma}^*_{S_1}) + \lambda\mathbf{I}_{S_1}\right)_{ii} = 0$ for $i \notin S_1$, and for all $i \in S_1$,

$$\left(q'_\lambda(\mathbf{\Gamma}^*_{S_1}) + \lambda\mathbf{I}_{S_1}\right)_{ii} = q'_\lambda\left((\boldsymbol{\gamma}^*)_i\right) + \lambda = p'_\lambda\left((\boldsymbol{\gamma}^*)_i\right) = 0,$$

where the last equality is because $p_\lambda(\cdot)$ satisfies the regularity condition (i) with $(\boldsymbol{\gamma}^*)_i \geq \nu$ for $i \in S_1$. Thus, we have $q'_\lambda(\mathbf{D}_{S_1}) + \lambda\mathbf{I}_{S_1} = \mathbf{0}$, which indicates that $\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{P}_\lambda(\mathbf{\Theta}^*)\right) = \mathbf{0}$. Therefore, we have

$$\begin{aligned}
\left\langle \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) + \lambda\mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\rangle &= \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{L}_n(\mathbf{\Theta}^*) + \nabla\mathcal{P}_\lambda(\mathbf{\Theta}^*)\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\rangle \\
&= \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\right), \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right\rangle \\
&\leq \left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 \cdot \left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right\|_*,
\end{aligned}$$

where the last inequality is derived from the Hölder inequality. What remains is to bound $\left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right\|_*$.

By the properties of projection on to the subspace $\mathcal{F}_{S_1}$, we have

$$\left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right\|_* \leq \sqrt{r_1}\left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right\|_F \leq \sqrt{r_1}\left\|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\|_F,$$

where the second inequality is due to the fact that $\operatorname{rank}\left(\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right)\right) \leq r_1$. Therefore, we have

$$\left\langle \mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) + \lambda\mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\rangle \leq \sqrt{r_1}\left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 \cdot \left\|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\|_F. \tag{B.4.7}$$

(3) Finally, consider $i \in S_2$ that $(\boldsymbol{\gamma}^*)_i \leq \nu$. Let $|S_2| = r_2$. Define a subspace of $\mathcal{F}$ associated with S_2 as follows

$$\mathcal{F}_{S_2}(\mathbf{U}^*, \mathbf{V}^*) := \left\{\mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2} | \operatorname{row}(\mathbf{\Delta}) \subset \mathbf{V}^*_{S_2} \text{ and } \operatorname{col}(\mathbf{\Delta}) \subset \mathbf{U}^*_{S_2}\right\},$$

where $\mathbf{U}^*_{S_2}$ and $\mathbf{V}^*_{S_2}$ is the matrix with the i^{th} row of $\mathbf{U}^*$ and $\mathbf{V}^*$ where $i \in S_2$. It is obvious that for all $\mathbf{\Delta} \in \mathbb{R}^{m_1 \times m_2}$, the following decomposition holds

$$\mathbf{\Pi}_{\mathcal{F}}(\mathbf{\Delta}) = \mathbf{\Pi}_{\mathcal{F}_{S_1}}(\mathbf{\Delta}) + \mathbf{\Pi}_{\mathcal{F}_{S_2}}(\mathbf{\Delta}).$$

In addition, since $\mathbf{U}^*$, $\mathbf{V}^*$ are unitary matrices, we have

$$\mathcal{F}_{S_1} \subset \mathcal{F}^{\perp}_{S_2}, \quad \text{and} \quad \mathcal{F}_{S_2} \subset \mathcal{F}^{\perp}_{S_1},$$

where $\mathcal{F}^{\perp}_{S_1}, \mathcal{F}^{\perp}_{S_2}$ denote the complementary subspace of $\mathcal{F}_{S_1}$ and $\mathcal{F}_{S_2}$, respectively. Similar to analysis in (2) on S_1, we have

$$\mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla\mathcal{Q}_\lambda(\mathbf{\Theta}^*)\right) = \mathbf{U}^*_{S_2} q'_\lambda(\mathbf{\Gamma}^*_{S_2})(\mathbf{V}^*_{S_2})^\top,$$

where $q'_\lambda(\mathbf{\Gamma}^*_{S_2})$ is a diagonal matrix that $\left(q'_\lambda(\mathbf{\Gamma}^*_{S_2})\right)_{ii} = 0$ for $i \notin S_2$, and for all $i \in S_2$, $\left(q'_\lambda(\mathbf{\Gamma}^*_{S_2})\right)_{ii} = q'_\lambda\left((\boldsymbol{\gamma}^*)_i\right) \leq \lambda$, since $(\boldsymbol{\gamma}^*)_i \leq \nu$ and $q_\lambda(\cdot)$ satisfies the regularity condition (iv). Therefore

$$\left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla\mathcal{Q}_\lambda(\mathbf{\Theta}^*)\right)\right\|_2 = \max_{i \in S_2}\left(q'_\lambda(\mathbf{\Gamma}^*_{S_2})\right)_{ii} \leq \lambda. \tag{B.4.8}$$

Meanwhile, we have

$$\left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\lambda \mathbf{W}^*)\right\|_2 \leq \left\|\mathbf{\Pi}_{\mathcal{F}}\left(\lambda \mathbf{U}^* \mathbf{V}^{*\top}\right)\right\|_2 = \lambda, \tag{B.4.9}$$

where the first inequality is due the fact that $\mathcal{F}_{S_2} \in \mathcal{F}$, and last equality comes from the fact that $\left\|\mathbf{U}^* \mathbf{V}^{*\top}\right\|_2 = 1$. Therefore, we have

$$\left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\lambda \mathbf{W}^*)\right\|_2 \leq \lambda. \tag{B.4.10}$$

In addition, we have the fact that $\left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 \leq \left\|\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right\|_2 \leq \lambda.$, which indicates that

$$\begin{aligned}
&\left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) + \lambda \mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle = \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*) + \nabla \mathcal{Q}_\lambda(\mathbf{\Theta}^*) + \lambda \mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle \\
&= \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle + \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{Q}_\lambda(\mathbf{\Theta}^*)\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle + \left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\lambda \mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle \\
&\leq \left[\left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 + \left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \mathcal{Q}_\lambda(\mathbf{\Theta}^*)\right)\right\|_2 + \left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\lambda \mathbf{W}^*)\right\|_2 \right] \left\|\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}})\right\|_*,
\end{aligned}$$

where the last inequality is due to Hölder's inequality. Since we have obtained the bound for each term, as in (B.4.8), (B.4.9), (B.4.10), we have

$$\begin{aligned}
\left\langle \mathbf{\Pi}_{\mathcal{F}_{S_2}}\left(\nabla \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) + \lambda \mathbf{W}^*\right), \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle &\leq 3\lambda \|\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}})\|_* \\
&\leq 3\lambda \sqrt{r_2} \|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\|_F,
\end{aligned} \tag{B.4.11}$$

where the last inequality utilizes the fact that $\operatorname{rank}(\mathbf{\Pi}_{\mathcal{F}_{S_2}}(\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}})) \leq r_2$.

Adding (B.4.6), (B.4.7), and (B.4.11), we have

$$\begin{aligned}
\left(\kappa(\mathfrak{X}) - \zeta_-\right) \|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F^2 &\leq \left\langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) + \lambda \mathbf{W}^*, \mathbf{\Theta}^* - \widehat{\mathbf{\Theta}} \right\rangle \\
&\leq \sqrt{r_1} \left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 \cdot \left\|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\right\|_F + 3\lambda \sqrt{r_2} \|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}\|_F,
\end{aligned}$$

which indicate that

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq \frac{\sqrt{r_1}}{\kappa(\mathfrak{X}) - \zeta_-} \left\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\left(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\right)\right\|_2 + \frac{3\lambda \sqrt{r_2}}{\kappa(\mathfrak{X}) - \zeta_-}.$$

This completes the proof. □

B.4.2 Proof of Theorem 4.2.5

Before presenting the proof of Theorem 4.2.5, we need the following lemma.

Lemma B.4.3 (Deterministic Bound)**.** *Suppose $\mathbf{\Theta}^* \in \mathbb{R}^{m_1 \times m_2}$ has rank r, $\mathfrak{X}(\cdot)$ satisfies RSC with respect to $\mathcal{C}$. Then the error bound between the oracle estimator $\widehat{\mathbf{\Theta}}_O$ and true $\mathbf{\Theta}^*$ satisfies*

$$\|\widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}^*\|_F \leq \frac{2\sqrt{r}\big\|\Pi_{\mathcal{F}}\big(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)\big)\big\|_2}{\kappa(\mathfrak{X})}, \tag{B.4.12}$$

Proof. Proof is provided in Section B.6.3. □

Proof of Theorem 4.2.5. Suppose $\widehat{\mathbf{W}} \in \partial\|\widehat{\mathbf{\Theta}}\|_*$, since $\widehat{\mathbf{\Theta}}$ is the solution to the SDP (4.1.2), the variational inequality yields

$$\max_{\mathbf{\Theta}'} \big\langle \widehat{\mathbf{\Theta}} - \mathbf{\Theta}', \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda \widehat{\mathbf{W}} \big\rangle \leq 0. \tag{B.4.13}$$

In the following, we will show that there exists some $\widehat{\mathbf{W}}_O \in \partial\|\widehat{\mathbf{\Theta}}_O\|_*$ such that, for all $\mathbf{\Theta}' \in \mathbb{R}^{m_1 \times m_2}$,

$$\max_{\mathbf{\Theta}'} \big\langle \widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}', \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O \big\rangle \leq 0. \tag{B.4.14}$$

Recall that $\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) = \mathcal{L}_n(\mathbf{\Theta}) + \mathcal{Q}_\lambda(\mathbf{\Theta})$. By projecting the components of the inner product of the LHS in (B.4.14) into two complementary spaces $\mathcal{F}$ and $\mathcal{F}^\perp$, we have the following decomposition

$$\begin{aligned} &\big\langle \widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}', \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O \big\rangle \\ &= \underbrace{\big\langle \mathbf{\Pi}_{\mathcal{F}}(\widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}'), \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O \big\rangle}_{I_1} + \underbrace{\big\langle \mathbf{\Pi}_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}'), \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O \big\rangle}_{I_2}. \end{aligned} \tag{B.4.15}$$

Analysis of Term I_1. Let $\boldsymbol{\gamma}^* = \boldsymbol{\gamma}(\mathbf{\Theta}^*)$, $\widehat{\boldsymbol{\gamma}}_O = \boldsymbol{\gamma}(\widehat{\mathbf{\Theta}}_O)$ be the vector of (ordered) singular values of $\mathbf{\Theta}^*$ and $\widehat{\mathbf{\Theta}}_O$, respectively. By the perturbation bounds for singular values, the Weyl's inequality [111], we have that

$$\max_i \big|(\boldsymbol{\gamma}^*)_i - (\widehat{\boldsymbol{\gamma}}_O)_i\big| \leq \big\|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}_O\big\|_2 \leq \big\|\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}_O\big\|_F.$$

Since Lemma B.4.3 provides the Frobenius norm on the estimation error $\mathbf{\Theta}^* - \widehat{\mathbf{\Theta}}_O$, we obtain that

$$\max_i \big|(\boldsymbol{\gamma}^*)_i - (\widehat{\boldsymbol{\gamma}}_O)_i\big| \leq \frac{2\sqrt{r}}{n\kappa(\mathfrak{X})}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2.$$

If it is assumed that $S = \mathrm{supp}(\boldsymbol{\sigma}^*)$, we have $|S| = r$. The triangle inequality yields that

$$\begin{aligned}\min_{i\in S}\big|(\widehat{\boldsymbol{\gamma}}_O)_i\big| &= \min_{i\in S}\big|(\widehat{\boldsymbol{\gamma}}_O)_i - (\boldsymbol{\gamma}^*)_i + (\boldsymbol{\gamma}^*)_i\big| \geq -\max_{i\in S}\big|(\widehat{\boldsymbol{\gamma}}_O - \boldsymbol{\gamma}^*)_i\big| + \min_{i\in S}\big|(\boldsymbol{\gamma}^*)_i\big| \\ &\geq -\frac{2\sqrt{r}}{n\kappa(\mathfrak{X})}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2 + \nu + \frac{2\sqrt{r}}{n\kappa(\mathfrak{X})}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2 \\ &= \nu,\end{aligned}$$

where the inequality on the second line is derived based on the condition that $\min_{i\in S}\big|(\boldsymbol{\gamma}^*)_i\big| \geq \nu + 2n^{-1}\sqrt{r}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_*/\kappa(\mathfrak{X})$. Based on the definition of oracle estimator (4.2.2), $\widehat{\boldsymbol{\Theta}}_O \in \mathcal{F}$, which implies $\mathrm{rank}(\widehat{\boldsymbol{\Theta}}_O) = r$. Therefore, we have

$$(\widehat{\boldsymbol{\gamma}}_O)_1 \geq (\widehat{\boldsymbol{\gamma}}_O)_2 \geq \ldots \geq (\widehat{\boldsymbol{\gamma}}_O)_r \geq \nu > 0 = (\widehat{\boldsymbol{\gamma}}_O)_{r+1} = (\widehat{\boldsymbol{\gamma}}_O)_m = 0. \tag{B.4.16}$$

By the definition of Oracle estimator, we have $\widehat{\boldsymbol{\Theta}}_O = \mathbf{U}^*\widehat{\boldsymbol{\Gamma}}_O\mathbf{V}^{*\top}$, where $\widehat{\boldsymbol{\Gamma}}_O$ is the diagonal matrix with $\mathrm{diag}(\widehat{\boldsymbol{\Gamma}}_O) = \widehat{\boldsymbol{\gamma}}_O$. Since $\mathcal{P}_\lambda(\boldsymbol{\Theta}) = \mathcal{Q}_\lambda(\boldsymbol{\Theta}) + \lambda\|\boldsymbol{\Theta}\|_*$, we have

$$\begin{aligned}\boldsymbol{\Pi}_{\mathcal{F}}\big(\nabla\mathcal{P}_\lambda(\widehat{\boldsymbol{\Theta}}_O)\big) &= \boldsymbol{\Pi}_{\mathcal{F}}\big(\nabla\mathcal{Q}_\lambda(\widehat{\boldsymbol{\Theta}}_O) + \lambda\partial\|\widehat{\boldsymbol{\Theta}}_O\|_*\big) \\ &= \boldsymbol{\Pi}_{\mathcal{F}}\big(\mathbf{U}^*q'_\lambda(\widehat{\boldsymbol{\Gamma}}_O)\mathbf{V}^{*\top} + \lambda\mathbf{U}^*\mathbf{V}^{*\top} + \lambda\widehat{\mathbf{Z}}_O\big) \\ &= \mathbf{U}^*\Big(q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)_S\big) + \lambda\mathbf{I}_r\Big)\mathbf{V}^{*\top},\end{aligned} \tag{B.4.17}$$

where $\widehat{\mathbf{Z}}_O \in \mathcal{F}^\perp$, $\|\widehat{\mathbf{Z}}_O\|_2 \leq 1$, and $(\widehat{\boldsymbol{\Gamma}}_O)_S \in \mathbb{R}^{r\times r}$ is a diagonal matrix with $\mathrm{diag}\big((\widehat{\boldsymbol{\Gamma}}_O)_S\big) = (\widehat{\boldsymbol{\gamma}}_O)_S$. The first equality in (B.4.17) is based on the definition of $\nabla\mathcal{Q}_\lambda(\cdot)$ and $\partial\|\cdot\|_*$, while the second is to simply project each component into the subspace $\mathcal{F}$. Since $p_\lambda(t) = q_\lambda(t) + \lambda|t|$, we have $p'_\lambda(t) = q'_\lambda(t) + \lambda t$ for all $t > 0$. Consider the diagonal matrix $q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)_S\big) + \lambda\mathbf{I}_r$, we have the i^{th} $(i \in S)$ element on the diagonal that

$$\Big(q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)_S\big) + \lambda\mathbf{I}_r\Big)_{ii} = q'_\lambda\big((\widehat{\boldsymbol{\gamma}}_O)_i\big) + \lambda = p'_\lambda\big((\widehat{\boldsymbol{\gamma}}_O)_i\big).$$

Since $p_\lambda(\cdot)$ satisfies the regularity condition (ii), that $p'_\lambda(t) = 0$ for all $t \geq \nu$, we have $p'_\lambda\big((\widehat{\boldsymbol{\gamma}}_O)_i\big) = 0$ for $i \in S$, in light of the fact that $(\widehat{\boldsymbol{\gamma}}_O)_i \geq \nu > 0$. Therefore, the diagonal matrix $q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)_S\big) + \lambda\mathbf{I}_r = \mathbf{0}$, substituting which into (B.4.17) yields

$$\boldsymbol{\Pi}_{\mathcal{F}}\big(\nabla\mathcal{P}_\lambda(\widehat{\boldsymbol{\Theta}}_O)\big) = \mathbf{0}. \tag{B.4.18}$$

Since $\widehat{\boldsymbol{\Theta}}_O$ is a minimizer of (4.2.2) over $\mathcal{F}$, we have the following optimality condition that for all $\boldsymbol{\Theta}' \in$

$\mathbb{R}^{m_1 \times m_2}$,

$$\max_{\boldsymbol{\Theta}'} \langle \boldsymbol{\Pi}_{\mathcal{F}}(\widehat{\boldsymbol{\Theta}}_O - \boldsymbol{\Theta}'), \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O) \rangle \leq 0. \tag{B.4.19}$$

Substitute (B.4.18) and (B.4.19) into item I_1, we have for all $\widehat{\mathbf{W}}_O \in \partial \|\widehat{\boldsymbol{\Theta}}_O\|_*$,

$$\begin{aligned} &\max_{\boldsymbol{\Theta}'} \langle \boldsymbol{\Pi}_{\mathcal{F}}(\widehat{\boldsymbol{\Theta}}_O - \boldsymbol{\Theta}'), \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O \rangle \\ &= \max_{\boldsymbol{\Theta}'} \langle \boldsymbol{\Pi}_{\mathcal{F}}(\widehat{\boldsymbol{\Theta}}_O - \boldsymbol{\Theta}'), \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O) \rangle + \max_{\boldsymbol{\Theta}'} \langle \boldsymbol{\Pi}_{\mathcal{F}}(\widehat{\boldsymbol{\Theta}}_O - \boldsymbol{\Theta}'), \boldsymbol{\Pi}_{\mathcal{F}}\big(\nabla \mathcal{P}_\lambda(\widehat{\boldsymbol{\Theta}}_O)\big) \rangle \\ &\leq 0. \end{aligned} \tag{B.4.20}$$

Analysis of Term I_2. By definition of $\nabla \mathcal{Q}_\lambda(\boldsymbol{\Theta})$, and the condition that $q'_\lambda(\cdot)$ satisfies the regularity condition (iii) in Assumption 4.2.3, we have the SVD of $\nabla \mathcal{Q}_\lambda(\boldsymbol{\Theta}_O)$ as $\nabla \mathcal{Q}_\lambda(\widehat{\boldsymbol{\Theta}}_O) = \mathbf{U}^* q'_\lambda(\widehat{\boldsymbol{\Gamma}}_O) \mathbf{V}^{*\top}$, where $\widehat{\boldsymbol{\Gamma}}_O \in \mathbb{R}^{r \times r}$ is a diagonal matrix. Projecting $\nabla \mathcal{Q}_\lambda(\widehat{\boldsymbol{\Theta}}_O)$ into $\mathcal{F}^\perp$ yields that

$$\begin{aligned} \boldsymbol{\Pi}_{\mathcal{F}^\perp}\big(\nabla \mathcal{Q}_\lambda(\widehat{\boldsymbol{\Theta}}_O)\big) &= \big(\mathbf{I}_{m_1} - \mathbf{U}^* \mathbf{U}^{*\top}\big) \mathbf{U}^* q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)\big) \mathbf{V}^{*\top} \big(\mathbf{I}_{m_1} - \mathbf{V}^* \mathbf{V}^{*\top}\big) \\ &= \big(\mathbf{U}^* - \mathbf{U}^*\big) q'_\lambda\big((\widehat{\boldsymbol{\Gamma}}_O)_{S^c}\big) \big(\mathbf{V}^{*\top} - \mathbf{V}^{*\top}\big) \\ &= \mathbf{0}. \end{aligned}$$

Thus,

$$\boldsymbol{\Pi}_{\mathcal{F}^\perp}\big(\nabla \mathcal{Q}_\lambda(\widehat{\boldsymbol{\Theta}}_O)\big) = \mathbf{0}. \tag{B.4.21}$$

Therefore,

$$I_2 = \big\langle \boldsymbol{\Pi}_{\mathcal{F}^\perp}(-\boldsymbol{\Theta}'), \boldsymbol{\Pi}_{\mathcal{F}^\perp}\big(\nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O\big) \big\rangle.$$

Moreover, the triangle inequality yields

$$\begin{aligned} \|\nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\|_2 &\leq \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*)\|_2 + \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*) - \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\|_2 \\ &\leq \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*)\|_2 + \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*) - \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\|_F \\ &\leq \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*)\|_2 + \rho(\mathfrak{X}) \|\boldsymbol{\Theta}^* - \widehat{\boldsymbol{\Theta}}_O\|_F, \end{aligned} \tag{B.4.22}$$

where the second inequality comes from the fact that $\|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*) - \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\|_2 \leq \|\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*) - \nabla \mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\|_F$,

while the last inequality is obtained by the restricted strong smoothness (Assumption 4.2.2), which is equivalent to

$$\|\nabla\mathcal{L}_n(\boldsymbol{\Theta}) - \nabla\mathcal{L}_n(\boldsymbol{\Theta} + \widehat{\boldsymbol{\Delta}}_O)\|_F \leq \rho(\mathfrak{X})\|\widehat{\boldsymbol{\Delta}}_O\|_F,$$

over the restricted set $\mathcal{C}$; since $\Pi_{\mathcal{F}^\perp}(\widehat{\boldsymbol{\Delta}}_O) = \mathbf{0}$, it is evident that $\widehat{\boldsymbol{\Delta}}_O \in \mathcal{C}$.

Substitute (B.4.12) of Lemma B.4.3 into (B.4.22), we have

$$\left\|\boldsymbol{\Pi}_{\mathcal{F}^\perp}\left(\nabla\mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\right)\right\|_2 \leq \left\|\nabla\mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\right\|_2 \leq \left\|\nabla\mathcal{L}_n(\boldsymbol{\Theta}^*)\right\|_2 + \frac{2\sqrt{r}\rho(\mathfrak{X})}{n\kappa(\mathfrak{X})}\left\|\mathfrak{X}^*(\boldsymbol{\epsilon})\right\|_2 \leq \lambda,$$

where the last inequality follows from the choice of λ.

By setting $\widehat{\mathbf{Z}}_O = -\lambda^{-1}\boldsymbol{\Pi}_{\mathcal{F}^\perp}\left(\nabla\mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O)\right)$, such that $\widehat{\mathbf{W}}_O = \mathbf{U}^*\mathbf{V}^{*\top} + \widehat{\mathbf{Z}}_O \in \partial\|\widehat{\boldsymbol{\Theta}}_O\|_*$ since $\widehat{\mathbf{Z}}_O$ satisfies the condition $\widehat{\mathbf{Z}}_O \in \mathcal{F}^\perp$, $\|\widehat{\mathbf{Z}}_O\|_2 \leq 1$, we have

$$\boldsymbol{\Pi}_{\mathcal{F}^\perp}\left(\nabla\mathcal{L}_n(\widehat{\boldsymbol{\Theta}}_O) + \lambda\widehat{\mathbf{W}}_O\right) = \mathbf{0},$$

which implies that

$$I_2 = \left\langle\boldsymbol{\Pi}_{\mathcal{F}^\perp}(-\boldsymbol{\Theta}'), \mathbf{0}\right\rangle = 0. \tag{B.4.23}$$

Substitute (B.4.20) and (B.4.23) into (B.4.15), we obtain (B.4.14) that

$$\max_{\boldsymbol{\Theta}'}\left\langle\widehat{\boldsymbol{\Theta}}_O - \boldsymbol{\Theta}', \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}_O) + \lambda\widehat{\mathbf{W}}_O\right\rangle \leq 0.$$

Now we are going to prove that $\widehat{\boldsymbol{\Theta}}_O = \boldsymbol{\Theta}^*$.

Applying Lemma B.4.1, we have

$$\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) \geq \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}_O) + \left\langle\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}_O), \widehat{\boldsymbol{\Theta}} - \widehat{\boldsymbol{\Theta}}_O\right\rangle + \frac{\kappa(\mathfrak{X}) - \zeta_-}{2}\|\widehat{\boldsymbol{\Theta}}_O - \widehat{\boldsymbol{\Theta}}\|_F^2, \tag{B.4.24}$$

$$\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}_O) \geq \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}) + \left\langle\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\boldsymbol{\Theta}}), \widehat{\boldsymbol{\Theta}}_O - \widehat{\boldsymbol{\Theta}}\right\rangle + \frac{\kappa(\mathfrak{X}) - \zeta_-}{2}\|\widehat{\boldsymbol{\Theta}}_O - \widehat{\boldsymbol{\Theta}}\|_F^2. \tag{B.4.25}$$

On the other hand, because of the convexity of nuclear norm $\|\cdot\|_*$, we obtain

$$\lambda\|\widehat{\boldsymbol{\Theta}}\|_* \geq \lambda\|\widehat{\boldsymbol{\Theta}}_O\|_* + \lambda\langle\widehat{\boldsymbol{\Theta}} - \widehat{\boldsymbol{\Theta}}_O, \widehat{\mathbf{W}}_O\rangle, \tag{B.4.26}$$

$$\lambda\|\widehat{\boldsymbol{\Theta}}_O\|_* \geq \lambda\|\widehat{\boldsymbol{\Theta}}\|_* + \lambda\langle\widehat{\boldsymbol{\Theta}}_O - \widehat{\boldsymbol{\Theta}}, \widehat{\mathbf{W}}\rangle. \tag{B.4.27}$$

Add (B.4.24) to (B.4.27), we obtain

$$0 \geq \underbrace{\langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda \widehat{\mathbf{W}}, \widehat{\mathbf{\Theta}}_O - \widehat{\mathbf{\Theta}} \rangle}_{I_3} + \underbrace{\langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O, \widehat{\mathbf{\Theta}} - \widehat{\mathbf{\Theta}}_O \rangle}_{I_4} + (\kappa(\mathfrak{X}) - \zeta_-) \|\widehat{\mathbf{\Theta}}_O - \widehat{\mathbf{\Theta}}\|_F^2. \quad \text{(B.4.28)}$$

Analysis of Term I_3. By (B.4.13), we have

$$\langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda \widehat{\mathbf{W}}, \widehat{\mathbf{\Theta}} - \widehat{\mathbf{\Theta}}_O \rangle \leq \max_{\mathbf{\Theta}'} \langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda \widehat{\mathbf{W}}, \widehat{\mathbf{\Theta}} - \mathbf{\Theta}' \rangle \leq 0. \quad \text{(B.4.29)}$$

Therefore $I_3 \geq 0$.

Analysis of Term I_4. By (B.4.14), we have

$$\langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O, \widehat{\mathbf{\Theta}}_O - \widehat{\mathbf{\Theta}} \rangle \leq \max_{\mathbf{\Theta}'} \langle \nabla \widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}_O) + \lambda \widehat{\mathbf{W}}_O, \widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}' \rangle \leq 0. \quad \text{(B.4.30)}$$

Therefore $I_4 \geq 0$. Substituting (B.4.29) and (B.4.30) into (B.4.28) yields that

$$(\kappa(\mathfrak{X}) - \zeta_-) \|\widehat{\mathbf{\Theta}}_O - \widehat{\mathbf{\Theta}}\|_F^2 \leq 0,$$

which holds if and only if

$$\widehat{\mathbf{\Theta}}_O = \widehat{\mathbf{\Theta}}, \quad \text{(B.4.31)}$$

because $\kappa(\mathfrak{X}) > \zeta_-$.

By Lemma B.4.3, we obtain the error bound

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F = \|\widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}^*\|_F \leq \frac{2\sqrt{r} \big\| \Pi_{\mathcal{F}}(\nabla \mathcal{L}_n(\mathbf{\Theta}^*)) \big\|_2}{\kappa(\mathfrak{X})},$$

which completes the proof. □

B.5 Proof of the Results for Specific Examples

In this section, we provide the detailed proofs for corollaries of specific examples presented in Section 4.2.2. We will first establish the RSC condition for both examples, followed by proofs of the corollaries and more results on oracle property respecting two specific examples of matrix completion.

Particularly, the proofs include the following components: (i) establish the RSC condition, obtaining

$\kappa(\mathfrak{X})$ by which Assumption 4.2.1 holds with high probability; (ii) estimate $\|\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\|_2$ for the choice of the regularity parameter λ; (iii) establish the RSS condition, obtaining $\rho(\mathfrak{X})$ by which Assumption 4.2.2 holds with high probability.

B.5.1 Matrix Completion

As shown in [18] with various examples, it is insufficient to recover the low-rank matrix, since it is infeasible to recover overly "spiky" matrices which have very few large entries. Some existing work [18] imposes stringent matrix incoherence conditions to preclude such matrices; these assumptions are relaxed in more recent work [72, 34] by restricting the spikiness ratio, which is defined as follows:

$$\alpha_{\mathrm{sp}}(\mathbf{\Theta}) = \frac{\sqrt{m_1 m_2}\|\mathbf{\Theta}\|_\infty}{\|\mathbf{\Theta}\|_F}.$$

Assumption B.5.1. *These exists a known α^*, such that*

$$\|\mathbf{\Theta}^*\|_\infty = \frac{\alpha_{\mathrm{sp}}(\mathbf{\Theta}^*)\|\mathbf{\Theta}^*\|_F}{\sqrt{m_1 m_2}} \leq \alpha^*.$$

For the example of matrix completion, we have the following matrix concentration inequality, which follows from Proof of Corollary 1 in [72].

Proposition B.5.2. *Let $\mathbf{X}_i$ uniformly distributed on $\mathcal{X}$, and $\{\xi_k\}_{k=1}^n$ be a finite sequence of independent Gaussian variables with variance σ^2. There exist constants C_1, C_2 that with probability at least $1 - C_2/M$, we have*

$$\Big\|\frac{1}{n}\sum_{i=1}^n \xi_i \mathbf{X}_i\Big\|_2 \leq C_1\sigma\sqrt{\frac{M\log M}{m_1 m_2 n}}.$$

Furthermore, the following Lemma plays a key rule in obtaining faster rate for estimator with nonconvex penalties. Particularly, the following Lemma will provide an upper bound on $\big\|\Pi_{\mathcal{F}}\big(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\big)\big\|_2$.

Lemma B.5.3. *If ξ_i is Gaussian noise with variance σ^2. $\mathcal{S}$ is a r-dimensional subspace. It holds with probability at least $1 - C_2/M$,*

$$\Big\|\Pi_{\mathcal{S}}\Big(\frac{1}{n}\sum_{i=1}^n \xi_i \mathbf{X}_i\Big)\Big\|_2 \leq C_1\sigma\sqrt{\frac{r\log M}{m_1 m_2 n}},$$

where C_1, C_2 are universal constants.

Proof. Proof is provided in Section B.6.4. □

In addition, we have the following Lemma (Theorem 1 in [72]), which plays central role in establishing the RSC condition.

Lemma B.5.4. *There are universal constants, $k_1, k_2, C_1, \ldots, C_5$, such that as long as $n > C_2 M \log M$, if the following condition is satisfied that*

$$\sqrt{m_1 m_2}\frac{\|\mathbf{\Delta}\|_\infty}{\|\mathbf{\Delta}\|_F}\frac{\|\mathbf{\Delta}\|_*}{\|\mathbf{\Delta}\|_F} \leq \frac{\sqrt{rn}}{k_1 r_1\sqrt{\log M} + k_2\sqrt{r_2 M \log M}}, \tag{B.5.1}$$

we have

$$\Big|\frac{\big\|\mathfrak{X}_n(\mathbf{\Delta})\big\|_2}{\sqrt{n}} - \frac{\|\mathbf{\Delta}\|_F}{\sqrt{m_1 m_2}}\Big| \leq \frac{7}{8}\frac{\|\mathbf{\Delta}\|_F}{\sqrt{m_1 m_2}}\Big[1 + \frac{C_1\alpha_{\mathrm{sp}}(\mathbf{\Delta})}{\sqrt{n}}\Big], \tag{B.5.2}$$

with probability greater than $1 - C_3\exp(-C_4 M\log M)$.

Proof of Corollary 4.2.6. With regard to the example of matrix completion, we consider a partially observed setting, *i.e.*, only the entries over the subset $\mathcal{X}$. A uniform sampling model is assumed that

$$\forall (i,j) \in \mathcal{X}, i \sim \mathrm{uniform}([m_1]), j \sim \mathrm{uniform}([m_2]).$$

Recall that $\widehat{\mathbf{\Delta}} = \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*$. In this proof, we consider two cases, depending on if the condition in (B.5.1) holds or not.

1. The condition in (B.5.1) does not hold.
2. The condition in (B.5.1) does hold.

Case 1. If the condition in (B.5.1) is violated, it implies that

$$\begin{aligned}
\|\widehat{\mathbf{\Delta}}\|_F^2 &\leq \sqrt{m_1 m_2}\|\widehat{\mathbf{\Delta}}\|_\infty \cdot \|\widehat{\mathbf{\Delta}}\|_* \frac{k_1 r_1\sqrt{\log M} + k_2\sqrt{r_2 M\log M}}{\sqrt{rn}} \\
&\leq \sqrt{m_1 m_2}(2\alpha^*)\big(\|\widehat{\mathbf{\Delta}}'\|_* + \|\widehat{\mathbf{\Delta}}''\|_*\big)\frac{k_1 r_1\sqrt{\log M} + k_2\sqrt{r_2 M\log M}}{\sqrt{rn}} \\
&\leq 12\alpha^*\sqrt{r m_1 m_2}\|\widehat{\mathbf{\Delta}}'\|_F \frac{k_1 r_1\sqrt{\log M} + k_2\sqrt{r_2 M\log M}}{\sqrt{rn}},
\end{aligned}$$

where $\widehat{\mathbf{\Delta}}' = \Pi_{\mathcal{F}}(\widehat{\mathbf{\Delta}})$ and $\widehat{\mathbf{\Delta}}'' = \Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Delta}})$, the second inequality follows from $\|\widehat{\mathbf{\Delta}}\|_\infty \leq \|\widehat{\mathbf{\Theta}}\|_\infty + \|\mathbf{\Theta}^*\|_\infty \leq 2\alpha^*$, and the decomposibility of nuclear norm that $\|\widehat{\mathbf{\Delta}}\|_* = \|\widehat{\mathbf{\Delta}}'\|_* + \|\widehat{\mathbf{\Delta}}''\|_*$; while the third inequality is based on the cone condition $\|\widehat{\mathbf{\Delta}}'\|_* \leq 5\|\widehat{\mathbf{\Delta}}''\|_*$ and $\|\widehat{\mathbf{\Delta}}'\|_* \leq \sqrt{r}\|\widehat{\mathbf{\Delta}}'\|_F$.

Moreover, since $\|\widehat{\mathbf{\Delta}}'\|_F \leq \|\widehat{\mathbf{\Delta}}\|_F$, we obtain that

$$\frac{1}{\sqrt{m_1 m_2}}\|\widehat{\mathbf{\Delta}}\|_F \leq 12\alpha^*\Big(k_1 r_1\sqrt{\frac{\log M}{n}} + k_1\sqrt{\frac{r_2 M\log M}{n}}\Big). \tag{B.5.3}$$

Case 2. The condition in (B.5.1) is satisfied.

As implied by (B.5.2), we have

$$\frac{\big\|\mathfrak{X}_n(\mathbf{\Delta})\big\|_2}{\sqrt{n}} \geq \frac{1}{8}\frac{\|\mathbf{\Delta}\|_F}{\sqrt{m_1 m_2}}\Big[1 - \frac{C_1'\alpha_{\mathrm{sp}}(\mathbf{\Delta})}{\sqrt{n}}\Big],$$

If $C_1'\alpha_{\mathrm{sp}}(\widehat{\mathbf{\Delta}})/\sqrt{n} > 1/2$, we have

$$\|\widehat{\mathbf{\Delta}}\|_F \leq 2C_2\sqrt{m_1 m_2}\frac{\|\widehat{\mathbf{\Delta}}\|_\infty}{\sqrt{n}} \leq 4C_2\alpha^*\sqrt{\frac{m_1 m_2}{n}}. \tag{B.5.4}$$

If $C_1'\alpha_{\mathrm{sp}}(\widehat{\mathbf{\Delta}})/\sqrt{n} \leq 1/2$, we have

$$\frac{\big\|\mathfrak{X}_n(\widehat{\mathbf{\Delta}})\big\|_2^2}{n} \geq \frac{C_6^2}{m_1 m_2}\|\widehat{\mathbf{\Delta}}\|_F^2. \tag{B.5.5}$$

In order to establish the RSC condition, we need to show that (B.5.5) is equivalent to Assumption 4.2.1.

$$\begin{aligned}
&\mathcal{L}_n(\mathbf{\Theta}^* + \widehat{\mathbf{\Delta}}) - \mathcal{L}_n(\mathbf{\Theta}^*) - \big\langle\nabla\mathcal{L}_n(\mathbf{\Theta}^*), \widehat{\mathbf{\Delta}}\big\rangle \\
&= \frac{1}{2n}\sum_{i=1}^n\big(\langle\mathbf{\Theta}^* + \widehat{\mathbf{\Delta}}, \mathbf{X}_i\rangle - y_i\big)^2 + \frac{1}{2n}\sum_{i=1}^n\big(\langle\mathbf{\Theta}^*, \mathbf{X}_i\rangle - y_i\big)^2 - \frac{1}{n}\sum_{i=1}^n\big(\langle\mathbf{\Theta}^*, \mathbf{X}_i\rangle - y_i\big)\langle\mathbf{X}_i, \widehat{\mathbf{\Delta}}\rangle \\
&= \frac{\big\|\mathfrak{X}_n(\widehat{\mathbf{\Delta}})\big\|_2^2}{n}.
\end{aligned}$$

Thus, we have that (B.5.5) establishes the RSC condition, and $\kappa(\mathfrak{X}) = C_6^2/(m_1 m_2)$.

After establishing the RSC condition, what remains is to upper bound $n^{-1}\big\|\mathfrak{X}^*(\boldsymbol{\epsilon})\big\|_2$ and $n^{-1}\big\|\Pi_{\mathcal{F}_{S_1}}\big(\mathfrak{X}^*(\boldsymbol{\epsilon})\big)\big\|_2$. By Proposition B.5.2, we have that with high probability,

$$\frac{1}{n}\big\|\mathfrak{X}^*(\boldsymbol{\epsilon})\big\|_2 \leq C_6\sigma\sqrt{\frac{M\log M}{m_1 m_2 n}}; \tag{B.5.6}$$

By Lemma B.5.3, we have that with high probability,

$$\frac{1}{n}\big\|\Pi_{\mathcal{F}_{S_1}}\big(\mathfrak{X}^*(\boldsymbol{\epsilon})\big)\big\|_2 \leq C_7\sigma\sqrt{\frac{r_1\log M}{m_1 m_2 n}}. \tag{B.5.7}$$

Substituting (B.5.6) and (B.5.7) into Theorem 4.2.4, we have that there exist positive constants C_1', C_2' such that

$$\frac{1}{\sqrt{m_1 m_2}}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq C_1' \sigma r_1 \sqrt{\frac{\log M}{n}} + C_2' \sigma \sqrt{\frac{r_2 M \log M}{n}}. \tag{B.5.8}$$

Putting pieces (B.5.3), (B.5.4), and (B.5.8) together, we have

$$\frac{1}{\sqrt{m_1 m_2}}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq \max\{\alpha^*, \sigma\}\left[C_3 r_1 \sqrt{\frac{\log M}{n}} + C_4 \sqrt{\frac{r_2 M \log M}{n}}\right],$$

which completes the proof. □

Corollary B.5.5. *Under the conditions of Theorem 4.2.5, suppose $\mathbf{X}_i$ uniformly distributed on $\mathcal{X}$. These exists positive constants $C_1, \ldots, C_4$, for any $t > 0$, if $\kappa(\mathfrak{X}) = C_1/(m_1 m_2) > \zeta_-$ and $\boldsymbol{\gamma}^*$ satisfies*

$$\min_{i \in S} \left|(\boldsymbol{\gamma}^*)_i\right| \geq \nu + C_2 \sigma \sqrt{r m_1 m_2} \sqrt{\frac{M \log M}{n}},$$

where $S = \operatorname{supp}(\boldsymbol{\sigma}^*)$, for estimator in (4.1.2) with regularization parameter

$$\lambda \geq C_3(1 + \sqrt{r})\sigma \sqrt{\frac{M \log M}{n m_1 m_2}},$$

we have that with high probability, $\widehat{\mathbf{\Theta}} = \widehat{\mathbf{\Theta}}_O$, which yields that $\operatorname{rank}(\widehat{\mathbf{\Theta}}) = \operatorname{rank}(\widehat{\mathbf{\Theta}}_O) = \operatorname{rank}(\mathbf{\Theta}^*) = r$. In addition, we have

$$\frac{1}{\sqrt{m_1 m_2}}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq C_4 r \sigma \sqrt{\frac{\log M}{n}}. \tag{B.5.9}$$

Proof of Corollary B.5.5. As shown in the proof of Corollary 4.2.6, we have $\kappa(\mathfrak{X}) = C_1/(m_1 m_2)$, together with (B.5.6) and (B.5.7), in order to prove Corollary B.5.5, according to Theorem 4.2.5, what remains is to obtain $\rho(\mathfrak{X})$ in Assumption 4.2.2. It can be shown that Assumption 4.2.2 is equivalent as

$$\frac{\rho(\mathfrak{X})}{2}\|\widehat{\mathbf{\Delta}}\|_F^2 \geq \frac{1}{n}\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2^2.$$

We consider the following cases depending on if (B.5.1) holds or not.

Case 1. If the condition in (B.5.1) is violated,

$$\frac{1}{n}\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_F^2 \leq \|\widehat{\mathbf{\Delta}}\|_\infty^2 \leq \|\widehat{\mathbf{\Delta}}\|_F^2,$$

which implies that $\rho(\mathfrak{X}) = 1$.

Case 2. The condition in (B.5.1) is satisfied. As implied by Lemma B.5.4, when $n \geq C_5^2 \alpha^* \geq C_5^2 \alpha_{\text{sp}}(\widehat{\mathbf{\Delta}})$, we have that with high probability, the following holds:

$$\frac{C_6}{m_1 m_2} \|\widehat{\mathbf{\Delta}}\|_F^2 \geq \frac{1}{n} \|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2^2.$$

Thus, $\rho(\mathfrak{X}) = C_6/(m_1 m_2)$, which completes the proof. □

B.5.2 Matrix Sensing With Dependent Sampling

In this subsection, we provide the proof for the results on matrix sensing. In particular, we will first establish the RSC condition for the application of matrix sensing, followed by the proof on faster convergence rate and more results on the oracle property.

In order to establish the RSC condition, we need the following lemma (Proposition 1 in [71]).

Lemma B.5.6. *Consider the sampling operator of $\mathbf{\Sigma}$-ensemble, it holds with probability at least $1 - 2\exp(-n/32)$ that*

$$\frac{\|\mathfrak{X}(\mathbf{\Delta})\|_2}{\sqrt{n}} \geq \frac{1}{4} \big\|\sqrt{\mathbf{\Sigma}}\text{vec}(\mathbf{\Delta})\big\|_2 - 12\pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big)\|\mathbf{\Delta}\|_*.$$

In addition, we need the upper bound of $n^{-1}\big\|\mathfrak{X}^*(\boldsymbol{\epsilon})\big\|_2$, as stated in the following Proposition (Lemma 6, [71]).

Proposition B.5.7. *With high probability, there are universal constants C_1, C_2 and C_3 such that*

$$\mathbb{P}\bigg[\frac{\big\|\mathfrak{X}^*(\boldsymbol{\epsilon})\big\|_2}{n} \geq C_1 \sigma \pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big)\bigg] \leq C_2 \exp\big(-C_3(m_1 + m_2)\big),$$

where $\pi(\mathbf{\Sigma})^2 = \sup_{\|\mathbf{u}\|_2=1, \|\mathbf{v}\|_2=1} \text{Var}(\mathbf{u}^\top \mathbf{X}\mathbf{v})$.

Proof of Corollary 4.2.8. To begin with, we need to establish the RSC condition as in Assumption 4.2.1. According to Lemma B.5.6, we have that

$$\frac{\big\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\big\|_2}{\sqrt{n}} \geq \frac{\sqrt{\lambda_{\min}(\mathbf{\Sigma})}}{4} \|\widehat{\mathbf{\Delta}}\|_F - 12\pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big)\|\widehat{\mathbf{\Delta}}\|_*.$$

By the decomposibility of nuclear norm, we have that

$$\|\widehat{\mathbf{\Delta}}\|_* = \|\widehat{\mathbf{\Delta}}'\|_* + \|\widehat{\mathbf{\Delta}}''\|_* \leq 6\|\widehat{\mathbf{\Delta}}'\|_* = 6\sqrt{r}\|\widehat{\mathbf{\Delta}}'\|_F \leq 6\sqrt{r}\|\widehat{\mathbf{\Delta}}\|_F, \tag{B.5.10}$$

where $\widehat{\mathbf{\Delta}}' = \Pi_{\mathcal{F}}(\widehat{\mathbf{\Delta}})$ and $\widehat{\mathbf{\Delta}}'' = \Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Delta}})$.

By substituting (B.5.10) into Proposition B.5.6, we have that

$$\begin{aligned}\frac{\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2}{\sqrt{n}} &\geq \frac{\sqrt{\lambda_{\min}(\mathbf{\Sigma})}}{4}\|\widehat{\mathbf{\Delta}}\|_F - 72\sqrt{r}\pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big)\|\widehat{\mathbf{\Delta}}\|_F \\ &= \Big[\frac{\sqrt{\lambda_{\min}(\mathbf{\Sigma})}}{4} - 72\sqrt{r}\pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big)\Big]\|\widehat{\mathbf{\Delta}}\|_F.\end{aligned}$$

Thus, for $n > C_1 r\pi^2(\mathbf{\Sigma})m_1m_2/\lambda_{\min}(\mathbf{\Sigma})$ where C_1 is sufficiently large such that

$$72\sqrt{r}\pi(\mathbf{\Sigma})\Big(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\Big) \leq \frac{\lambda_{\min}(\mathbf{\Sigma})}{8},$$

we have

$$\frac{\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2}{\sqrt{n}} \geq \frac{\sqrt{\lambda_{\min}(\mathbf{\Sigma})}}{8}\|\widehat{\mathbf{\Delta}}\|_F,$$

which implies that

$$\frac{\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2^2}{n} \geq \frac{\lambda_{\min}(\mathbf{\Sigma})}{64}\|\widehat{\mathbf{\Delta}}\|_F^2.$$

Therefore, $\kappa(\mathfrak{X}) = \lambda_{\min}(\mathbf{\Sigma})/32$ such that the following holds,

$$\frac{\|\mathfrak{X}(\widehat{\mathbf{\Delta}})\|_2^2}{n} \geq \frac{\kappa(\mathfrak{X})}{2}\|\widehat{\mathbf{\Delta}}\|_F^2,$$

which establishes the RSC condition for matrix sensing.

On the other hand, we have

$$\big\|\mathbf{\Pi}_{\mathcal{F}_{S_1}}\big(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\big)\big\|_2 = \big\|\mathbf{U}_{S_1}^*\mathbf{U}_{S_1}^{*\top}\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\mathbf{V}_{S_1}^*\mathbf{V}_{S_1}^{*\top}\big\|_2 = \big\|\mathbf{U}_{S_1}^{*\top}\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\mathbf{V}_{S_1}^*\big\|_2,$$

where the second inequality follows from the property of left and right singular vectors $\mathbf{U}_{S_1}^*, \mathbf{V}_{S_1}^*$.

It is worth noting that $\mathbf{U}_{S_1}^{*\top}\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\mathbf{V}_{S_1}^* \in \mathbb{R}^{r_1\times r_1}$. By Proposition B.5.7, we have that

$$\begin{aligned}\big\|\mathbf{U}^{*\top}\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\mathbf{V}^*\big\|_2 &\leq 2C_0\sigma\pi(\mathbf{\Sigma})\sqrt{\frac{M}{n}}, \\ \big\|\mathbf{U}_{S_1}^{*\top}\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\mathbf{V}_{S_1}^*\big\|_2 &\leq 2C_0\sigma\pi(\mathbf{\Sigma})\sqrt{\frac{r_1}{n}},\end{aligned} \tag{B.5.11}$$

which hold with probability at lease $1 - C_1\exp(-C_2 r_1)$.

The upper bound is obtained directed from Theorem 4.2.4 and (B.5.11). Thus, we complete the proof. □

Corollary B.5.8. *Under the condition of Theorem 4.2.5, for some universal constants $C_1, \ldots, C_6$ if $\kappa(\mathfrak{X}) = C_1\lambda_{\min}(\mathbf{\Sigma}) > \zeta_-$ and $\boldsymbol{\gamma}^*$ satisfies*

$$\min_{i\in S} \big|(\boldsymbol{\gamma}^*)_i\big| \geq \nu + C_2\sigma\pi(\mathbf{\Sigma})\frac{\sqrt{r}(\sqrt{m_1}+\sqrt{m_2})}{\sqrt{n}\lambda_{\min}(\mathbf{\Sigma})},$$

where $S = \mathrm{supp}(\boldsymbol{\gamma}^*)$, for estimator in (4.1.2) with regularization parameter

$$\lambda \geq C_3\big(1 + \frac{\sqrt{r}\lambda_{\max}(\mathbf{\Sigma})}{\lambda_{\min}(\mathbf{\Sigma})}\big)\sigma\pi(\mathbf{\Sigma})\bigg(\sqrt{\frac{m_1}{n}} + \sqrt{\frac{m_2}{n}}\bigg),$$

we have that $\widehat{\mathbf{\Theta}} = \widehat{\mathbf{\Theta}}_O$, which yields that $rank(\widehat{\mathbf{\Theta}}) = rank(\widehat{\mathbf{\Theta}}_O) = rank(\mathbf{\Theta}^*) = r$, with probability at least $1 - C_4\exp\big(-C_5(m_1+m_2)\big)$. In addition, we have

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\|_F \leq \frac{C_6 r\pi(\mathbf{\Sigma})}{\sqrt{n}\lambda_{\min}(\mathbf{\Sigma})}. \tag{B.5.12}$$

Proof of Corollary B.5.8. The proof follows from the proof of Corollary 4.2.8 and Theorem 4.2.5. As shown in the proof of Corollary 4.2.8, we have $\kappa(\mathfrak{X}) = C_1\lambda_{\min}(\mathbf{\Sigma})$, together with (B.5.11), in order to prove Corollary B.5.8, according to Theorem 4.2.5, what remains is to obtain $\rho(\mathfrak{X})$ in Assumption 4.2.2, respecting the example of matrix sensing.

According to Assumption 4.2.2, we have that $\rho(\mathfrak{X}) = \lambda_{\max}(\mathbf{H}_n)$, where $\mathbf{H}_n$ is the Hessian matrix of $\mathcal{L}_n(\cdot)$. Based on the definition of $\mathcal{L}_n(\cdot)$, we have

$$\mathbf{H}_n = n^{-1}\sum_{i=1}^{n}\mathrm{vec}(\mathbf{X}_i)\mathrm{vec}(\mathbf{X}_i)^\top.$$

Thus $\mathbb{E}[\mathbf{H}_n] = \mathbf{\Sigma}$. By concentration, we have that when n is sufficiently large, with high probability, $\lambda_{\max}(\mathbf{H}_n) \leq 2\lambda_{\max}(\mathbf{\Sigma})$, which is equivalent to $\rho(\mathfrak{X}) \leq 2\lambda_{\max}(\mathbf{\Sigma})$, holding with high probability, where n is sufficiently large. This completes the proof. □

B.6 Proof of Auxiliary Lemmas

In this section, we present the proof of auxiliary lemmas.

B.6.1 Proof of Lemma B.4.1

Proof. By the restricted strong convexity assumption (Assumption 4.2.1), we have

$$\mathcal{L}_n(\boldsymbol{\Theta}_2) \geq \mathcal{L}_n(\boldsymbol{\Theta}_1) + \langle \nabla \mathcal{L}_n(\boldsymbol{\Theta}_1), \boldsymbol{\Theta}_2 - \boldsymbol{\Theta}_1 \rangle + \frac{\kappa(\mathfrak{X})}{2} \|\boldsymbol{\Theta}_2 - \boldsymbol{\Theta}_1\|_F^2. \tag{B.6.1}$$

In the following, we will show the strong smoothness of $\mathcal{Q}_\lambda(\cdot)$, based on the regularity condition (ii), which imposes constraint on the level of nonconvexity of $q_\lambda(\cdot)$. Assume $\boldsymbol{\gamma}_1 = \boldsymbol{\gamma}(\boldsymbol{\Theta}_1), \boldsymbol{\gamma}_2 = \boldsymbol{\gamma}(\boldsymbol{\Theta}_2)$ are the vectors of singular values of $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2$, respectively, and the singular values in $\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2$ are nonincreasing. For $\boldsymbol{\Theta}_1$, $\boldsymbol{\Theta}_2$, we have the following singular value decompositions:

$$\boldsymbol{\Theta}_1 = \mathbf{U}_1 \boldsymbol{\Gamma}_1 \mathbf{V}_1^\top,$$
$$\boldsymbol{\Theta}_2 = \mathbf{U}_2 \boldsymbol{\Gamma}_2 \mathbf{V}_2^\top,$$

where $\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2 \in \mathbb{R}^{m \times m}$ are diagonal matrix with $\boldsymbol{\Gamma}_1 = \operatorname{diag}(\boldsymbol{\gamma}_1), \boldsymbol{\Gamma}_2 = \operatorname{diag}(\boldsymbol{\gamma}_2)$. For each pair of singular values of $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2$: $\big((\boldsymbol{\gamma}_1)_i, (\boldsymbol{\gamma}_2)_i\big)$ where $i = 1, 2, \ldots, m$, we have

$$-\zeta_- \big((\boldsymbol{\gamma}_1)_i - (\boldsymbol{\gamma}_2)_i\big)^2 \leq \big[q_\lambda'\big((\boldsymbol{\gamma}_1)_i\big) - q_\lambda'\big((\boldsymbol{\gamma}_2)_i\big)\big]\big((\boldsymbol{\gamma}_1)_i - (\boldsymbol{\gamma}_2)_i\big),$$

which is equivalent to

$$\big\langle \big(-q_\lambda'(\boldsymbol{\Gamma}_1)\big) - \big(-q_\lambda'(\boldsymbol{\Gamma}_2)\big), \boldsymbol{\Gamma}_1 - \boldsymbol{\Gamma}_2 \big\rangle \leq \zeta_- \|\boldsymbol{\Gamma}_1 - \boldsymbol{\Gamma}_2\|_F^2,$$

which yields

$$\big\langle \big(-\nabla \mathcal{Q}_\lambda(\boldsymbol{\Theta}_1)\big) - \big(-\nabla \mathcal{Q}_\lambda(\boldsymbol{\Theta}_2)\big), \boldsymbol{\Theta}_1 - \boldsymbol{\Theta}_2 \big\rangle \leq \zeta_- \|\boldsymbol{\Theta}_1 - \boldsymbol{\Theta}_2\|_F^2. \tag{B.6.2}$$

Since (B.6.2) is the definition of strongly smoothness of $-\mathcal{Q}(\cdot)$, it can be show to be equivalent to the following inequality that

$$\mathcal{Q}_\lambda(\boldsymbol{\Theta}_2) \geq \mathcal{Q}_\lambda(\boldsymbol{\Theta}_1) + \langle \nabla \mathcal{Q}(\boldsymbol{\Theta}_1), \boldsymbol{\Theta}_2 - \boldsymbol{\Theta}_1) - \frac{\zeta_-}{2} \|\boldsymbol{\Theta}_2 - \boldsymbol{\Theta}_1\|_F^2. \tag{B.6.3}$$

Adding up (B.6.1) and (B.6.3), we complete the proof. □

B.6.2 Proof of Lemma B.4.2

Proof. By Lemma B.4.1, we have that

$$\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda\|\widehat{\mathbf{\Theta}}\|_* - \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*) - \lambda\|\mathbf{\Theta}^*\|_* \geq \langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\rangle + \lambda\|\widehat{\mathbf{\Theta}}\|_* - \lambda\|\mathbf{\Theta}^*\|_*. \tag{B.6.4}$$

For the first term on the RHS in (B.6.4), we have the following lower bound

$$\begin{aligned}\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\rangle &= \big\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\rangle + \big\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\rangle \\ &\geq -\underbrace{\big\|\Pi_{\mathcal{F}}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*)\big)\big\|_2}_{I_1}\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* \\ &\quad - \underbrace{\big\|\Pi_{\mathcal{F}^\perp}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*)\big)\big\|_2}_{I_2}\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_*,\end{aligned} \tag{B.6.5}$$

where the last inequality follows from Hölder's inequality.

Analysis of term I_1. It can be shown that $\nabla\mathcal{L}_n(\mathbf{\Theta}^*) = -\mathfrak{X}^*(\boldsymbol{\epsilon})/n$. Based on the condition that $\lambda > 2n^{-1}\|\mathfrak{X}^*(\boldsymbol{\epsilon})\|_2$, we have that

$$\|\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\|_2 \leq \lambda/2. \tag{B.6.6}$$

Moreover, by condition (iv) in Assumption 4.2.3 and (B.6.6), we obtain that

$$\big\|\Pi_{\mathcal{F}}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*)\big)\big\|_2 = \big\|\Pi_{\mathcal{F}}\big(\nabla\mathcal{L}_n(\mathbf{\Theta}^*) + \mathcal{Q}_\lambda(\mathbf{\Theta}^*)\big)\big\|_2 \leq 3\lambda/2.$$

Analysis of term I_2. Since $\Pi_{\mathcal{F}^\perp}(\mathbf{\Theta}^*) = \mathbf{0}$, we have that

$$\big\|\Pi_{\mathcal{F}^\perp}\big(\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*)\big)\big\|_2 = \big\|\Pi_{\mathcal{F}^\perp}\big(\nabla\mathcal{L}_n(\mathbf{\Theta}^*)\big)\big\|_2 \leq \lambda/2. \tag{B.6.7}$$

Putting pieces (B.6.6) and (B.6.7) into (B.6.5), we obtain

$$\langle \nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\rangle \geq -3\lambda/2\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* - \lambda/2\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_*. \tag{B.6.8}$$

Meanwhile, we have the lower bound on $\lambda\|\widehat{\mathbf{\Theta}}\|_* - \lambda\|\mathbf{\Theta}\|_*$ that

$$\begin{aligned}\lambda\|\widehat{\mathbf{\Theta}}\|_* - \lambda\|\mathbf{\Theta}\|_* &= \lambda\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}})\big\|_* + \lambda\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}})\big\|_* - \lambda\|\mathbf{\Theta}\|_* \\ &\geq -\lambda\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* + \lambda\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* \end{aligned} \tag{B.6.9}$$

Adding (B.6.8) and (B.6.9) yields that

$$\langle\nabla\widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}^*), \widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*\rangle + \lambda\|\widehat{\mathbf{\Theta}}\|_* - \lambda\|\mathbf{\Theta}\|_* = -5\lambda/2\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* + \lambda/2\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_*. \tag{B.6.10}$$

Due to the fact that $\widehat{\mathbf{\Theta}}$ is the global minimizer of (4.1.2), provided the condition that $\kappa(\mathfrak{X}) > \zeta_-$, we have

$$\widetilde{\mathcal{L}}_{n,\lambda}(\widehat{\mathbf{\Theta}}) + \lambda\|\widehat{\mathbf{\Theta}}\|_* - \widetilde{\mathcal{L}}_{n,\lambda}(\mathbf{\Theta}) - \lambda\|\mathbf{\Theta}^*\|_* \leq 0. \tag{B.6.11}$$

Substituting (B.6.10) and (B.6.11) into (B.6.4), since $\lambda > 0$, we have that

$$\big\|\Pi_{\mathcal{F}^\perp}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_* \leq 5\big\|\Pi_{\mathcal{F}}(\widehat{\mathbf{\Theta}} - \mathbf{\Theta}^*)\big\|_*,$$

which completes the proof. □

B.6.3 Proof of Lemma B.4.3

Proof. $\widehat{\mathbf{\Delta}}_O = \widehat{\mathbf{\Theta}}_O - \mathbf{\Theta}^*$. According to observation model (4.1.1) and definition of $\mathfrak{X}(\cdot)$, we have

$$\begin{aligned}\mathcal{L}_n(\widehat{\mathbf{\Theta}}_O) - \mathcal{L}_n(\mathbf{\Theta}^*) &= \frac{1}{2n}\sum_{i=1}^n \big(y_i - \mathfrak{X}_i(\mathbf{\Theta}^* + \widehat{\mathbf{\Delta}}_O)\big)^2 - \frac{1}{2n}\sum_{i=1}^n \big(y_i - \mathfrak{X}_i(\mathbf{\Theta}^*)\big)^2 \\ &= \frac{1}{2n}\sum_{i=1}^n \big(\epsilon_i - \mathfrak{X}_i(\widehat{\mathbf{\Delta}}_O)\big)^2 - \frac{1}{2n}\sum_{i=1}^n \epsilon_i^2 \\ &= \frac{1}{2n}\|\mathfrak{X}(\widehat{\mathbf{\Delta}}_O)\|_2^2 - \frac{1}{n}\langle\mathfrak{X}^*(\boldsymbol{\epsilon}), \widehat{\mathbf{\Delta}}_O\rangle,\end{aligned}$$

where $\mathfrak{X}^*(\boldsymbol{\epsilon}) = \sum_{i=1}^n \epsilon_i\mathbf{X}_i$ is the adjoint of the operator $\mathfrak{X}$. Because the oracle estimator $\widehat{\mathbf{\Theta}}_O$ minimizes $\mathcal{L}_n(\cdot)$ over the subspace $\mathcal{F}$, while $\mathbf{\Theta}^* \in \mathcal{F}$, we have $\mathcal{L}_n(\widehat{\mathbf{\Theta}}_O) - \mathcal{L}_n(\mathbf{\Theta}^*) \leq 0$, which yields

$$\frac{1}{2n}\|\mathfrak{X}(\widehat{\mathbf{\Delta}}_O)\|_2^2 \leq \frac{1}{n}\langle\mathfrak{X}^*(\boldsymbol{\epsilon}), \widehat{\mathbf{\Delta}}_O\rangle. \tag{B.6.12}$$

On the other hand, recall that by the RSC condition (Assumption 4.2.1), we have

$$\mathcal{L}_n(\boldsymbol{\Theta}+\boldsymbol{\Delta}) \geq \mathcal{L}_n(\boldsymbol{\Theta}) + \langle \nabla \mathcal{L}_n(\boldsymbol{\Theta}), \boldsymbol{\Delta} \rangle + \kappa(\mathfrak{X})/2 \|\boldsymbol{\Delta}\|_F^2,$$

which implies that

$$\frac{1}{2n}\|\mathfrak{X}(\widehat{\boldsymbol{\Delta}}_O)\|_2^2 - \frac{1}{n}\langle \mathfrak{X}^*(\boldsymbol{\epsilon}), \widehat{\boldsymbol{\Delta}}_O \rangle - \langle \nabla \mathcal{L}_n(\boldsymbol{\Theta}^*), \boldsymbol{\Delta} \rangle = \frac{1}{2n}\|\mathfrak{X}(\widehat{\boldsymbol{\Delta}}_O)\|_2^2 \geq \frac{\kappa(\mathfrak{X})}{2}\|\widehat{\boldsymbol{\Delta}}_O\|_F^2. \tag{B.6.13}$$

Substituting (B.6.13) into (B.6.12), we have

$$\frac{\kappa(\mathfrak{X})}{2}\|\widehat{\boldsymbol{\Delta}}_O\|_F^2 \leq \frac{1}{2n}\|\mathfrak{X}(\widehat{\boldsymbol{\Delta}}_O)\|_2^2 \leq \frac{1}{n}\langle \mathfrak{X}^*(\boldsymbol{\epsilon}), \widehat{\boldsymbol{\Delta}}_O \rangle. \tag{B.6.14}$$

Therefore,

$$\|\widehat{\boldsymbol{\Delta}}_O\|_F^2 \leq \frac{2\langle \Pi_{\mathcal{F}}(\mathfrak{X}^*(\boldsymbol{\epsilon})), \widehat{\boldsymbol{\Delta}}_O \rangle}{n\kappa(\mathfrak{X})} \leq \frac{2\big\|\Pi_{\mathcal{F}}(\mathfrak{X}^*(\boldsymbol{\epsilon}))\big\|_2 \|\widehat{\boldsymbol{\Delta}}_O\|_*}{n\kappa(\mathfrak{X})},$$

where the last inequality is due to Hölder inequality. Moreover, since the rank $\boldsymbol{\Delta}_O$ is r, we have the fact that $\|\widehat{\boldsymbol{\Delta}}_O\|_* \leq \sqrt{r}\|\widehat{\boldsymbol{\Delta}}_O\|_F$, which indicates that

$$\|\widehat{\boldsymbol{\Delta}}_O\|_F^2 \leq \frac{2\sqrt{r}\big\|\Pi_{\mathcal{F}}(\mathfrak{X}^*(\boldsymbol{\epsilon}))\big\|_2 \cdot \|\widehat{\boldsymbol{\Delta}}_O\|_F}{n\kappa(\mathfrak{X})}.$$

Therefore, we have the following deterministic error bound

$$\|\widehat{\boldsymbol{\Delta}}_O\|_F \leq \frac{2\sqrt{r}\big\|\Pi_{\mathcal{F}}(\mathfrak{X}^*(\boldsymbol{\epsilon}))\big\|_2}{n\kappa(\mathfrak{X})} = \frac{2\sqrt{r}\big\|\Pi_{\mathcal{F}}(\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*))\big\|_2}{\kappa(\mathfrak{X})},$$

where the last equality results from the fact that $\nabla \mathcal{L}_n(\boldsymbol{\Theta}^*) = -\mathfrak{X}^*(\boldsymbol{\epsilon})/n$.

Thus, we complete the proof. □

B.6.4 Proof of Lemma B.5.3

In order to prove Lemma B.5.3, we need the Ahlswede-Winter Matrix Bound. To begin with, we introduce the definition of $\|\cdot\|_{\psi_1}$ and $\|\cdot\|_{\psi_2}$, followed by some established results on $\|\cdot\|_{\psi_1}$ and $\|\cdot\|_{\psi_2}$.

The sub-Gaussian norm of X, denoted by $\|X\|_{\psi_2}$, is defined as follows

$$\|X\|_{\psi_2} = \sup_{p \geq 1} p^{-1/2}(\mathbb{E}|X|^p)^{1/p}.$$

It is known that if $\mathbb{E}[X] = 0$, then $\mathbb{E}[\exp(tX)] \leq \exp(Ct^2\|X\|_{\psi_2}^2)$ for all $t \in \mathbb{R}$.

The sub-Exponential norm of X, denoted by $\|X\|_{\psi_1}$, is defined as follows

$$\|X\|_{\psi_1} = \sup_{p \geq 1} p^{-1}(\mathbb{E}|X|^p)^{1/p}.$$

By [103], we have the following Lemma.

Lemma B.6.1. *For Z_1 and Z_2 being two sub-Gaussian random variables, $Z_1 Z_2$ is a sub-exponential random variable with*

$$\|Z_1 Z_2\|_{\psi_1} \leq C \max\big\{\|Z_1\|_{\psi_2}^2,\ \|Z_2\|_{\psi_2}^2\big\},$$

where $C > 0$ is an absolute constant.

Theorem B.6.2 (Ahlswede-Winter Matrix Bound)**.** *[72] Let $\mathbf{Z}_1, \ldots, \mathbf{Z}_n$ be random matrices of size $m_1 \times m_2$. Let $\|\mathbf{Z}_i\|_{\psi_1} \leq K$ for all i such that $\|\mathbf{Z}_i\|_{\psi_1}$ is upper bounded by K. Furthermore, we have $\delta_i^2 = \max\big\{\big\|\mathbb{E}[\mathbf{Z}_i^\top \mathbf{Z}_i]\big\|_2, \big\|\mathbb{E}[\mathbf{Z}_i \mathbf{Z}_i^\top]\big\|_2\big\}$, and $\delta^2 = \sum_{i=1}^n \delta_i^2$. Then we have*

$$\mathbb{P}\bigg(\Big\|\sum_{i=1}^n \mathbf{Z}_i\Big\|_2 \geq t\bigg) \leq m_1 m_2 \max\Big\{\exp\Big(-\frac{t^2}{4\delta^2}\Big), \exp\Big(-\frac{t}{2K}\Big)\Big\}.$$

Now we are ready to prove Lemma B.5.3.

Proof of Lemma B.5.3. Since $\mathbf{U}^*$ and $\mathbf{V}^*$ are singular vectors, for $\mathcal{S} = \mathcal{F}(\mathbf{U}^*, \mathbf{V}^*)$, we have

$$\begin{aligned}\frac{1}{n}\Big\|\Pi_{\mathcal{S}}\Big(\sum_{i=1}^n \xi_i \mathbf{X}_i\Big)\Big\|_2 &= \frac{1}{n}\Big\|\mathbf{U}^*\mathbf{U}^{*\top}\Big(\sum_{i=1}^n \xi_i \mathbf{X}_i\Big)\mathbf{V}^*\mathbf{V}^{*\top}\Big\|_2 \\ &= \frac{1}{n}\Big\|\mathbf{U}^{*\top}\Big(\sum_{i=1}^n \xi_i \mathbf{X}_i\Big)\mathbf{V}^*\Big\|_2.\end{aligned}$$

Recall that $\mathbf{X}_i = \mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top$. Let $\mathbf{Y}_i = \epsilon_i \mathbf{X}_i = \epsilon_i \mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top$. We have $\|\mathbf{Y}_i\|_{\psi_1} \leq C\sigma^2$. Let $\mathbf{Z}_i = \mathbf{U}^{*\top}\mathbf{Y}_i\mathbf{V}^* \in \mathbb{R}^{r\times r}$. We have

$$\|\mathbf{Z}_i\|_{\psi_1} = \big\|\mathbf{U}^{*\top}\mathbf{Y}_i\mathbf{V}^*\big\|_{\psi_1}.$$

Based on the definition of $\mathbf{Y}_i$, we have that $\|\mathbf{Z}_i\|_{\psi_1} < C\sigma$. By applying Theorem B.6.1, we have

$$\|\mathbf{Z}_i\|_{\psi_1} \leq C'\sigma^2.$$

Thus, $K = C'\sigma^2$.

Furthermore, we have

$$\begin{aligned}\mathbb{E}[\mathbf{Z}_i\mathbf{Z}_i^\top] &= \mathbb{E}[\mathbf{U}^{*\top}\mathbf{Y}_i\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{Y}_i^\top\mathbf{U}^*] = \mathbb{E}[\epsilon_i^2\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*] \\ &= \sigma^2\mathbb{E}[\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*]\end{aligned}$$

Based on the definition of spectral norm, we have

$$\begin{aligned}\left\|\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*\right\|_2 &= \max_{\|\mathbf{a}\|_2=1}\mathbf{a}^\top\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*\mathbf{a} \\ &= \max_{\|\mathbf{b}\|_2=1}\mathbf{b}^\top\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{b},\end{aligned}$$

where the second equality follows by setting $\mathbf{b} = \mathbf{U}^*\mathbf{a} \in \mathbb{R}^{m_1}$. In addition, we have

$$\mathbf{b}^\top\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{b} = \mathbf{b}_{j(i)}\mathbf{v}_k^*\mathbf{v}_k^{*\top}\mathbf{b}_{j(i)} = \mathbf{b}_{j(i)}^2\|\mathbf{v}_k^*\|_2^2,$$

where $\mathbf{v}_k^*$ is the k-th row of $\mathbf{V}^*$. Thus

$$\begin{aligned}\left\|\mathbb{E}[\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*]\right\|_2 &= \left\|\frac{1}{m_1m_2}\sum_{j=1}^{m_1}\sum_{k=2}^{m_2}\mathbf{U}^{*\top}\mathbf{e}_j\mathbf{e}_k^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_k\mathbf{e}_j^\top\mathbf{U}^*\right\|_2 \\ &= \frac{1}{m_1m_2}\max_{\|\mathbf{a}\|_2=1}\mathbf{a}^\top\sum_{j=1}^{m_1}\sum_{k=2}^{m_2}\mathbf{U}^{*\top}\mathbf{e}_j\mathbf{e}_k^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_k\mathbf{e}_j^\top\mathbf{U}^*\mathbf{a} \\ &= \frac{1}{m_1m_2}\max_{\|\mathbf{b}\|_2=1}\sum_{j=1}^{m_1}\sum_{k=2}^{m_2}b_j^2\|\mathbf{v}_k^*\|_2^2.\end{aligned}$$

Since $\sum_{j=1}^{m_1} b_j^2 = 1$ and $\sum_{k=1}^{m_2}\|\mathbf{v}_k^*\|_2^2 = \|\mathbf{V}^*\|_F^2 = r$, we obtain that

$$\left\|\mathbb{E}[\mathbf{U}^{*\top}\mathbf{e}_{j(i)}\mathbf{e}_{k(i)}^\top\mathbf{V}^*\mathbf{V}^{*\top}\mathbf{e}_{k(i)}\mathbf{e}_{j(i)}^\top\mathbf{U}^*]\right\|_2 = \frac{r}{m_1m_2}.$$

Therefore, we have

$$\left\|\mathbb{E}[\mathbf{Z}_i\mathbf{Z}_i^\top]\right\|_2 = \frac{\sigma^2 r}{m_1m_2},$$

and the same result also applies to $\left\|\mathbb{E}[\mathbf{Z}_i^\top\mathbf{Z}_i]\right\|_2$.

By applying Theorem B.6.2, we obtain that

$$\mathbb{P}\Big(\Big\|\sum_{i=1}^{n}\xi_i\mathbf{Z}_i\Big\|_2 \geq t\Big) \leq m_1m_2\max\Big\{\exp\Big(-\frac{m_1m_2t^2}{4n\sigma^2r}\Big), \exp(-\frac{t}{2\sigma^2})\Big\}.$$

Thus, with probability at least $1 - C_2M^{-1}$, we have

$$\Big\|\sum_{i=1}^{n}\xi_i\mathbf{Z}_i\Big\|_2 \leq C_1\sigma\sqrt{\frac{nr\log M}{m_1m_2}}$$

where $M = \max(m_1, m_2)$. It immediately implies that

$$\Big\|\frac{1}{n}\sum_{i=1}^{n}\xi_i\mathbf{Z}_i\Big\|_2 \leq C_1\sigma\sqrt{\frac{r\log M}{m_1m_2n}}, \tag{B.6.15}$$

which completes the proof. □